

INSTITUTE
OF
HYDROLOGY

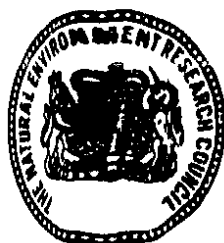
MICROCOMPUTER PROGRAMS TO ASSIST
THE ACQUISITION AND INTERPRETATION
RESISTIVITY SOUNDING DATA

by

J W FINCH

ABSTRACT

This report describes and lists four programs, designed for use on microcomputers, for the acquisition and interpretation of resistivity sounding data. The programs are designed to be used principally with the BGS Offset Wenner system and are written for the Epson HX-20 and Research Machines 360Z microcomputers. They are user-friendly and actions by the operator are in response to prompts from the program in a question-and-answer form.



REPORT NO 91

September 1984

CONTENTS

	Page
INTRODUCTION	1
HARDWARE REQUIREMENTS	1
SYSTEMS DESIGN	1
BGS OFFSET WENNER SYSTEM	2
PROGRAM 1	3
PROGRAM 2	7
PROGRAM 3	8
PROGRAM 4	16
REFERENCES	33

INTRODUCTION

This report describes four programs that have been designed to assist the acquisition and interpretation of vertical electrical sounding data. The first three programs are designed for use with data acquired using the BGS Offset Wenner sounding system. The fourth program is more general and can accept any data acquired using either Wenner or Schlumberger data.

The programs are designed to be user-friendly and data entries or decisions are made by the operator in response to prompts using question and answer. All entries are in free format and are echoed by the program to be verified by the operator. The first two programs, used for data acquisition and transfer, are written in Basic for use on the Epson HX-20 portable computer. The other programs, for data acquisition and interpretation, are in Microsoft Fortran for the Research Machines 380Z.

HARDWARE REQUIREMENTS

The first two programs require a standard Epson HX-20 portable computer fitted with a microcassette drive, plus a cable with the appropriate connectors to link the Epson HX-20 to the Research Machines 380Z.

The other two programs require a Research Machines 380Z microcomputer with 56K RAM, a high resolution graphics board, twin double density, dual sided, 5.25" floppy disc drives, a colour monitor and an Epson FX-80 dot-matrix printer. In addition, Microsoft Fortran is necessary as is some form of software capable of editing text, programs and data.

SYSTEMS DESIGN

The basic concept of the system is to use the HX-20 in the field for data acquisition and the 380Z in the office for interactive, iterative interpretation. Data is entered using the HX-20's keyboard in response to prompts and, after verification, it is stored on a microcassette and listed on the microprinter. This is done using program 1. Program 2 is then used to transfer data from microcassettes to disc files on the 380Z. To do this it is necessary to modify one of the utility routines that run under CP/M, the 380Z's operating system. Program 3 runs on the 380Z and processes the data from the HX-20 to produce Wenner sounding curves which it stores in disc files. These can then be interpreted using Program 4.

The program requirements are:

	Language	Size
1	Basic	4946 bytes
2	Basic	2492 bytes
3	Fortran	22460 bytes
4	Fortran	33843 bytes

The soundings are identified by their reference number. This can have up to eight characters which may be alphanumeric or special characters such as - or /. It is important that the reference number is unique and so it is recommended that a systematic approach to the numbers is adopted. Combinations of map codes and dates are often effective.

On the 3802 the filename qualifier .DAT is used to signify unprocessed BGS readings whilst the qualifier .INT signifies unformatted apparent resistivity curves.

BGS OFFSET WENNER SYSTEM

This field system allows Wenner apparent resistivity curve data to be acquired more accurately and with less man power than conventional systems. The equipment consists of two multicore cables fitted with metallic take-outs at pre-determined distances, a number of electrodes, a switch box and cables to connect the system to a resistivity meter.

The system uses a five electrode array with the electrodes equi-spaced. However, only four electrodes are used at a time. Thus, for each electrode spacing a series of measurements can be made using different electrode combinations. Two of these configurations D1 and D2 are conventional Wenner arrays but shifted by one spacing. Barker (1980) has shown that by taking the mean of these two readings a substantial reduction in the effects of lateral inhomogeneities can be achieved. The three remaining configurations A, B and C do not use the central electrodes but are a tripotential system. They allow interpolation between the points produced by the offset Wenner readings.

The use of a multicore cable reduces the time taken to lay out the cables and move the electrodes. The fixed takeouts for the electrodes reduce errors due to incorrect spacing distances whilst the system of readings allow further checks on data quality to be carried out.

Three measures of error are possible from the readings. The first, the reading error, is a measure solely of the instrument, operator and electrode array. The sum of readings B and C must equal reading A, irrespective of the subsurface conditions. Thus any significant deviation from this equality indicates a malfunction in the system.

The second measure is that if subsurface conditions are laterally homogeneous, the two readings D1 and D2 will be equal. Hence the degree of difference between the two readings is a measure of subsurface lateral inhomogeneity. This is called the offset error.

The third error measure, the potential ladder error, is generated when the sets of A, B and C readings are used to interpolate between the offset data. This error is also a measure of subsurface lateral inhomogeneity.

Program 1 in this report allows these error measures to be calculated in the field so that quality checks can be made when the data is acquired. It also allows the full Wenner resistivity curve to be generated so that a field interpretation can be carried out. This allows subsequent survey work to be planned on the basis of the data already collected.

PROGRAM 1

This program is designed to be used in the field to acquire BGS Offset Wenner data. It should be loaded and run at the beginning of the day's work and then the HX-20 should be left on until the day's work is completed. This is because when the program is started, it creates a file on the microcassette which it names with the date. Thus, if the program is run on the same day with the same tape it will create two files with the same name.

The program uses the RAM file of the HX-20 to keep a record of the files that occur on the microcassette. Before a microcassette is removed from the HX-20 this list of files is written on to a file named CASO:DIR.DIR at the beginning of the tape. Before a microcassette can be used a blank copy of this file must be written on to it. This is done using a program called CASSFMT. Microcassettes that contain old data that is no longer required can be re-used after the directory file has been rewritten as blank.

For each sounding, the program asks initially for data about the site of the sounding. Unknown information can be entered as blank or zero. The information requested is:

Site grid reference
Azimuth of the array
The V.E.S. (site) number

The operator is then asked to enter readings. For each set of readings he is asked initially for the electrode spacing number, as marked on the BGS switch box. If this spacing has already been used then the operator is asked to verify that it is to be used again. This protects against accidental repetition of readings but allows readings to be repeated if a mistake was made. The operator is then asked to give the configuration he is using, i.e. whether it is A, B, C, D1 or D2, and the reading. If the configuration has already been used then a warning is given and the operator is asked to verify the request.

When the five readings have been given for an electrode spacing the program checks the reading error and, if it exceeds 1%, a warning is displayed and the operator is asked to verify that set of readings. The offset error is also given so that a running check on data quality is made.

When the operator has given the readings for all the electrode spacings he wishes to use the program checks that the spacings used were those intended. If this is verified then the input readings, the calculated error measures and Wenner apparent resistivity curve are listed on the microprinter. The input readings are then written to the microcassette. A listing of the output from this program is given below.

File CASE=09/27/83.DAT
starting at 2

U.E.S. FP9

Grid Ref. 61472 89812
Azimuth 13

INPUT DATA

A	C	B
13.380	14.700	0.6040
7.550	6.910	0.5220
4.870	4.540	0.3050
2.590	2.410	0.1700
1.090	1.030	0.0500
0.346	0.323	0.0210

D1	D2
11.100	11.600
5.970	5.370
3.510	3.430
1.880	2.120
0.850	0.080
0.275	0.201

ERROR MEASURES

Obs.	Offset	Pot. Lad
-0.0033	-0.841	0.0037
0.0136	0.1050	-0.1270
0.0051	0.0231	0.0400
0.0079	-0.1200	-0.0220
0.0018	-0.0437	0.0025
0.0010	-0.0216	0.0000

PROCESSING RESULTS

Spac.	Resis.
0.5	35.7
1.0	35.6
1.5	41.5
2.0	43.6
3.0	47.5
4.0	50.3
6.0	49.0
8.0	43.7
12.0	34.8
16.0	27.9
24.0	22.3
32.0	18.5

Mean obs. err. 0.0077
Mean off. err. 0.0712
Mean pot. err. 0.0002

```

10 CLEAR200,480:DEFFIL16,0:BX=0:FX=0:FD$=SPACE$(12):FORI=0TO29:PUTXI,SX,FX,FD$:N
EXTI:GOTO100
20 IE=0:IFID$="00"THENGOTO30ELSEIFID$="10"THENGOTO40ELSEIFID$="CF"THENGOTO80ELSE
IFID$="0F"THENGOTO50ELSEIFID$="1F"THENGOTO90ELSERETURN
30 WIND-4000:TAPCNT=0:OPEN"0",E1,"CASO:DIR.DIR":FORI=0TO29:GETXI,SX,FX,FD$:PRINT
E1,SX,FX,FD$:NEXTI:CLOSEE1:WIND-4000:TAPCNT=0:RETURN
40 WIND-4000:TAPCNT=0:OPEN"1",E1,"CASO:DIR.DIR":FORI=0TO29:IFEOF(1)THENCLOSEE1:W
IND-4000:TAPCNT=0:RETURNELSEINPUTE1,SX,FX,FD$:PUTXI,SX,FX,FD$:NEXTI:CLOSEE1:WIND
-4000:TAPCNT=0:RETURN
50 IFLEN(FD$)<12THENFD$=FD$+SPACE$(8-LEN(FD$))+".DAT"
60 WIND-4000:TAPCNT=0:FORI=0TO29:GETXI,SX,FX,FF$:IF$X=<0THENIFI=0THENSX=200:GOTO
70ELSEGETXI(I-1),FX,SX,FF$:GOTO70ELSENEXTI:IE=1:RETURN
70 IT=I:BX=SX+5:WIND$X=OPEN"0",E1,("CASO:"+FD$):RETURN
80 CLOSEE1:FX=TAPCNT:PUTXTT,SX,FX,FD$:WIND-4000:TAPCNT=0:RETURN
90 IFLEN(FD$)<12THENFD$=FD$+SPACE$(8-LEN(FD$))+".DAT":WIND-4000:TAPCNT=0:FORI=0T
O29:GETXI,SX,FX,FF$:IFF$=FD$THENWIND(SX-5):OPEN"1",E1,("CASO:"+FD$):RETURNELSEI
F$X=0THENIE=1:RETURNELSENEXTI:IE=1:RETURN
100 DIM R(5,11),ES(5),E1(11),E2(11),E3(11),AA(25),RW(25),D(11),DD(11)
110 PI=8.0*ATN(1.0)
120 PRINT "Case. ready Y/N"
130 AS=INKEY$:IF AS="Y" THEN 150 ELSE IF AS="N" THEN140 ELSE 130
140 PRINT "Insert cassette":PRINT "and start again":END
150 ID$="ID":GOSUB20:FD$=DATE$+".DAT":ID$="OF":GOSUB20
160 IFIE=1THENPRINT"CASSETTE FULL":GOTO120
170 LPRINT: LPRINT"File ";FD$
180 FOR I=0 TO 24:AA(I)=0.0: NEXT I
190 DATA A,C,D1,D2,B
200 FOR I=0 TO 4: READ ES(I): NEXT I
210 FOR I=0 TO 4: FOR J=0 TO 9: R(I,J)=0.0: NEXT J: NEXT I
220 CLS:PRINT "Give grid letters":PRINT "of the grid ref.":INPUT$J$
230 PRINT "Give easting of"
240 INPUT "v.e.s.":B1
250 PRINT "Give northing of"
260 INPUT "v.e.s.":B2
270 CLS:PRINT "Grid reference": PRINT SJ$:B1:B2
280 PRINT "Correct Y/N"
290 AS=INKEY$:IF AS="Y" THEN 300 ELSE IF AS="N" THEN 220 ELSE 290
300 CLS:PRINT "Give azimuth of"
310 INPUT "array":AZ
320 CLS:PRINT "Azimuth =" :AZ
330 PRINT "Correct Y/N"
340 AS=INKEY$:IF AS="Y" THEN 350 ELSE IF AS="N" THEN 300 ELSE 340
350 CLS:PRINT "Give v.e.s."
360 INPUT "number":V$
370 CLS:PRINT "V.e.s. =" :V$
380 PRINT "Correct Y/N"
390 AS=INKEY$:IF AS="Y" THEN 400 ELSE IF AS="N" THEN 350 ELSE 390
400 CLS:PRINT "Give electrode spa."
410 INPUT "number":N
420 CLS:PRINT "Spacing no. =" :N
430 PRINT "Correct Y/N"
440 AS=INKEY$:IF AS="Y" THEN 450 ELSE IF AS="N" THEN 400 ELSE 440
450 FOR K=0 TO 4: IF R(K,N)<>0 THEN 470: ELSE NEXT K
460 GOTO 500
470 CLS:SOUND 4,4: PRINT "Spacing already used"
480 PRINT "replace Y/N"
490 AS=INKEY$:IF AS="Y" THEN 500 ELSE IF AS="N" THEN 400 ELSE 490
500 FOR K=0 TO 4
510 CLS:PRINT "Which config."
520 INPUT "A,C,D1,D2 or B":C$
530 FOR I=0 TO 4: IF ES(I)=C$ THEN 550 ELSE NEXT I
540 CLS:PRINT "No such config.": GOTO 510
550 IF R(I,N)=0.0 THEN 580 ELSE SOUND 4,4: PRINT "reading already given"
560 PRINT "Replace Y/N"
570 AS=INKEY$:IF AS="Y" THEN 580 ELSE IF AS="N" THEN 510 ELSE 570
580 INPUT "Give reading": R(I,N)
590 CLS:PRINT "Config. =" :C$
600 PRINT "Reading =" : R(I,N)
610 PRINT "Correct Y/N"
620 AS=INKEY$:IFA$="Y" THEN 630 ELSE IF AS="N" THEN 550 ELSE 620
630 NEXT K
640 E1(N)=(R(0,N)-R(1,N)-R(2,N))/5(0,N)
650 CLS:PRINT "Obs. error =" :PRINT USING"###.###":E1(N)
660 IF ABS(E1(N))<0.01 THEN 680 ELSE PRINT "Obs. error > 0.01": PRINT "please ve
rify":PRINT "Correct Y/N"
670 AS=INKEY$:IF AS="Y" THEN 680 ELSE IF AS="N" THEN 450 ELSE 670
680 E2(N)=2*(R(2,N)-R(3,N))/(1/2(N)+R(3,N))

```

```

690 PRINT "Offset err =";PRINT USING"###.###":E2(N)
700 PRINT "Another spacing":PRINT"Y/N"
710 AS=INKEY$:IF AS="Y" THEN 400 ELSE IF AS="N" THEN 720 ELSE 710
720 NS=-1: NF=0
730 FOR I=0 TO 9
740 FOR J=0 TO 4
750 IF R(J,I)<>0 THEN IF NS=-1 THEN NS=I
760 IF R(J,I)=0 THEN IF NS<-1 THEN NF=I-1: GOTO 780
770 NEXT J: NEXT I
780 CLS:PRINT "Spacings used":PRINT "from ";NS:PRINT "to ";NF
790 PRINT "Correct Y/N"
800 AS=INKEY$:IF AS="Y" THEN 810 ELSE IF AS="N" THEN 400 ELSE 800
810 LPRINT:LPRINT:LPRINT "V.E.S. ";V$
820 AA(NS)=0.5*NS
830 FOR I=NS TO NF
840 J=(I-NS+1)*2+NS-1:K=J+1
850 AA(J)=AA(NS)*2^(I-NS+1)
860 AA(K)=3.0*AA(NS)*2^(I-NS)
870 NEXT I
880 LPRINT:LPRINT"Grid Ref. ";:LPRINT USING"###.###":G1:LPRINT USING"###.###":G2
890 LPRINT " A="A: " B="B: " C="C: " D="D: " E="E: " F="F: " G="G: " H="H: " I="I: " J="J: " K="K: " L="L: " M="M: " N="N: " O="O: " P="P: " Q="Q: " R="R: " S="S: " T="T: " U="U: " V="V: " W="W: " X="X: " Y="Y: " Z="Z: "
900 LPRINT:LPRINT"INPUT DATA"
910 LPRINT:LPRINT" A. C B:LE INT
920 FOR I=NS TO NF
930 LPRINT USING"###.###":R(0,I):LPRINT USING"###.###":R(1,I):LPRINT USING"###.###":R(2,I):LPRINT USING"###.###":R(3,I):LPRINT USING"###.###":R(4,I)
940 NEXT I
950 LPRINT:LPRINT" D1 D2":LPRINT
960 ES1=0: ES2=0: ES3=0
970 PRINT:G1,G2,G3,G4,G5,G6,G7,G8,G9,G10,G11,G12,G13,G14,G15,G16,G17,G18,G19,G20,G21,G22,G23,G24,G25,G26,G27,G28,G29,G30,G31,G32,G33,G34,G35,G36,G37,G38,G39,G40,G41,G42,G43,G44,G45,G46,G47,G48,G49,G50,G51,G52,G53,G54,G55,G56,G57,G58,G59,G60,G61,G62,G63,G64,G65,G66,G67,G68,G69,G70,G71,G72,G73,G74,G75,G76,G77,G78,G79,G80,G81,G82,G83,G84,G85,G86,G87,G88,G89,G90,G91,G92,G93,G94,G95,G96,G97,G98,G99,G100,G101,G102,G103,G104,G105,G106,G107,G108,G109,G110,G111,G112,G113,G114,G115,G116,G117,G118,G119,G120,G121,G122,G123,G124,G125,G126,G127,G128,G129,G130,G131,G132,G133,G134,G135,G136,G137,G138,G139,G140,G141,G142,G143,G144,G145,G146,G147,G148,G149,G150,G151,G152,G153,G154,G155,G156,G157,G158,G159,G160,G161,G162,G163,G164,G165,G166,G167,G168,G169,G170,G171,G172,G173,G174,G175,G176,G177,G178,G179,G180,G181,G182,G183,G184,G185,G186,G187,G188,G189,G190,G191,G192,G193,G194,G195,G196,G197,G198,G199,G200,G201,G202,G203,G204,G205,G206,G207,G208,G209,G210,G211,G212,G213,G214,G215,G216,G217,G218,G219,G220,G221,G222,G223,G224,G225,G226,G227,G228,G229,G230,G231,G232,G233,G234,G235,G236,G237,G238,G239,G240,G241,G242,G243,G244,G245,G246,G247,G248,G249,G250,G251,G252,G253,G254,G255,G256,G257,G258,G259,G260,G261,G262,G263,G264,G265,G266,G267,G268,G269,G270,G271,G272,G273,G274,G275,G276,G277,G278,G279,G280,G281,G282,G283,G284,G285,G286,G287,G288,G289,G290,G291,G292,G293,G294,G295,G296,G297,G298,G299,G300,G301,G302,G303,G304,G305,G306,G307,G308,G309,G310,G311,G312,G313,G314,G315,G316,G317,G318,G319,G320,G321,G322,G323,G324,G325,G326,G327,G328,G329,G330,G331,G332,G333,G334,G335,G336,G337,G338,G339,G340,G341,G342,G343,G344,G345,G346,G347,G348,G349,G350,G351,G352,G353,G354,G355,G356,G357,G358,G359,G360,G361,G362,G363,G364,G365,G366,G367,G368,G369,G370,G371,G372,G373,G374,G375,G376,G377,G378,G379,G380,G381,G382,G383,G384,G385,G386,G387,G388,G389,G390,G391,G392,G393,G394,G395,G396,G397,G398,G399,G400,G401,G402,G403,G404,G405,G406,G407,G408,G409,G410,G411,G412,G413,G414,G415,G416,G417,G418,G419,G420,G421,G422,G423,G424,G425,G426,G427,G428,G429,G430,G431,G432,G433,G434,G435,G436,G437,G438,G439,G440,G441,G442,G443,G444,G445,G446,G447,G448,G449,G450,G451,G452,G453,G454,G455,G456,G457,G458,G459,G460,G461,G462,G463,G464,G465,G466,G467,G468,G469,G470,G471,G472,G473,G474,G475,G476,G477,G478,G479,G480,G481,G482,G483,G484,G485,G486,G487,G488,G489,G490,G491,G492,G493,G494,G495,G496,G497,G498,G499,G500,G501,G502,G503,G504,G505,G506,G507,G508,G509,G510,G511,G512,G513,G514,G515,G516,G517,G518,G519,G520,G521,G522,G523,G524,G525,G526,G527,G528,G529,G530,G531,G532,G533,G534,G535,G536,G537,G538,G539,G540,G541,G542,G543,G544,G545,G546,G547,G548,G549,G550,G551,G552,G553,G554,G555,G556,G557,G558,G559,G560,G561,G562,G563,G564,G565,G566,G567,G568,G569,G570,G571,G572,G573,G574,G575,G576,G577,G578,G579,G580,G581,G582,G583,G584,G585,G586,G587,G588,G589,G590,G591,G592,G593,G594,G595,G596,G597,G598,G599,G600,G601,G602,G603,G604,G605,G606,G607,G608,G609,G610,G611,G612,G613,G614,G615,G616,G617,G618,G619,G620,G621,G622,G623,G624,G625,G626,G627,G628,G629,G630,G631,G632,G633,G634,G635,G636,G637,G638,G639,G640,G641,G642,G643,G644,G645,G646,G647,G648,G649,G650,G651,G652,G653,G654,G655,G656,G657,G658,G659,G660,G661,G662,G663,G664,G665,G666,G667,G668,G669,G670,G671,G672,G673,G674,G675,G676,G677,G678,G679,G680,G681,G682,G683,G684,G685,G686,G687,G688,G689,G690,G691,G692,G693,G694,G695,G696,G697,G698,G699,G700,G701,G702,G703,G704,G705,G706,G707,G708,G709,G710,G711,G712,G713,G714,G715,G716,G717,G718,G719,G720,G721,G722,G723,G724,G725,G726,G727,G728,G729,G730,G731,G732,G733,G734,G735,G736,G737,G738,G739,G740,G741,G742,G743,G744,G745,G746,G747,G748,G749,G750,G751,G752,G753,G754,G755,G756,G757,G758,G759,G760,G761,G762,G763,G764,G765,G766,G767,G768,G769,G770,G771,G772,G773,G774,G775,G776,G777,G778,G779,G780,G781,G782,G783,G784,G785,G786,G787,G788,G789,G790,G791,G792,G793,G794,G795,G796,G797,G798,G799,G800,G801,G802,G803,G804,G805,G806,G807,G808,G809,G810,G811,G812,G813,G814,G815,G816,G817,G818,G819,G820,G821,G822,G823,G824,G825,G826,G827,G828,G829,G830,G831,G832,G833,G834,G835,G836,G837,G838,G839,G840,G841,G842,G843,G844,G845,G846,G847,G848,G849,G850,G851,G852,G853,G854,G855,G856,G857,G858,G859,G860,G861,G862,G863,G864,G865,G866,G867,G868,G869,G870,G871,G872,G873,G874,G875,G876,G877,G878,G879,G880,G881,G882,G883,G884,G885,G886,G887,G888,G889,G890,G891,G892,G893,G894,G895,G896,G897,G898,G899,G900,G901,G902,G903,G904,G905,G906,G907,G908,G909,G910,G911,G912,G913,G914,G915,G916,G917,G918,G919,G920,G921,G922,G923,G924,G925,G926,G927,G928,G929,G930,G931,G932,G933,G934,G935,G936,G937,G938,G939,G940,G941,G942,G943,G944,G945,G946,G947,G948,G949,G950,G951,G952,G953,G954,G955,G956,G957,G958,G959,G960,G961,G962,G963,G964,G965,G966,G967,G968,G969,G970,G971,G972,G973,G974,G975,G976,G977,G978,G979,G980,G981,G982,G983,G984,G985,G986,G987,G988,G989,G990,G991,G992,G993,G994,G995,G996,G997,G998,G999,G1000,G1001,G1002,G1003,G1004,G1005,G1006,G1007,G1008,G1009,G1010,G1011,G1012,G1013,G1014,G1015,G1016,G1017,G1018,G1019,G1020,G1021,G1022,G1023,G1024,G1025,G1026,G1027,G1028,G1029,G1030,G1031,G1032,G1033,G1034,G1035,G1036,G1037,G1038,G1039,G1040,G1041,G1042,G1043,G1044,G1045,G1046,G1047,G1048,G1049,G1050,G1051,G1052,G1053,G1054,G1055,G1056,G1057,G1058,G1059,G1060,G1061,G1062,G1063,G1064,G1065,G1066,G1067,G1068,G1069,G1070,G1071,G1072,G1073,G1074,G1075,G1076,G1077,G1078,G1079,G1080,G1081,G1082,G1083,G1084,G1085,G1086,G1087,G1088,G1089,G1090,G1091,G1092,G1093,G1094,G1095,G1096,G1097,G1098,G1099,G1100,G1101,G1102,G1103,G1104,G1105,G1106,G1107,G1108,G1109,G1110,G1111,G1112,G1113,G1114,G1115,G1116,G1117,G1118,G1119,G1120,G1121,G1122,G1123,G1124,G1125,G1126,G1127,G1128,G1129,G1130,G1131,G1132,G1133,G1134,G1135,G1136,G1137,G1138,G1139,G1140,G1141,G1142,G1143,G1144,G1145,G1146,G1147,G1148,G1149,G1150,G1151,G1152,G1153,G1154,G1155,G1156,G1157,G1158,G1159,G1160,G1161,G1162,G1163,G1164,G1165,G1166,G1167,G1168,G1169,G1170,G1171,G1172,G1173,G1174,G1175,G1176,G1177,G1178,G1179,G1180,G1181,G1182,G1183,G1184,G1185,G1186,G1187,G1188,G1189,G1190,G1191,G1192,G1193,G1194,G1195,G1196,G1197,G1198,G1199,G1200,G1201,G1202,G1203,G1204,G1205,G1206,G1207,G1208,G1209,G1210,G1211,G1212,G1213,G1214,G1215,G1216,G1217,G1218,G1219,G1220,G1221,G1222,G1223,G1224,G1225,G1226,G1227,G1228,G1229,G1230,G1231,G1232,G1233,G1234,G1235,G1236,G1237,G1238,G1239,G1240,G1241,G1242,G1243,G1244,G1245,G1246,G1247,G1248,G1249,G1250,G1251,G1252,G1253,G1254,G1255,G1256,G1257,G1258,G1259,G1260,G1261,G1262,G1263,G1264,G1265,G1266,G1267,G1268,G1269,G1270,G1271,G1272,G1273,G1274,G1275,G1276,G1277,G1278,G1279,G1280,G1281,G1282,G1283,G1284,G1285,G1286,G1287,G1288,G1289,G1290,G1291,G1292,G1293,G1294,G1295,G1296,G1297,G1298,G1299,G1300,G1301,G1302,G1303,G1304,G1305,G1306,G1307,G1308,G1309,G1310,G1311,G1312,G1313,G1314,G1315,G1316,G1317,G1318,G1319,G1320,G1321,G1322,G1323,G1324,G1325,G1326,G1327,G1328,G1329,G1330,G1331,G1332,G1333,G1334,G1335,G1336,G1337,G1338,G1339,G1340,G1341,G1342,G1343,G1344,G1345,G1346,G1347,G1348,G1349,G1350,G1351,G1352,G1353,G1354,G1355,G1356,G1357,G1358,G1359,G1360,G1361,G1362,G1363,G1364,G1365,G1366,G1367,G1368,G1369,G1370,G1371,G1372,G1373,G1374,G1375,G1376,G1377,G1378,G1379,G1380,G1381,G1382,G1383,G1384,G1385,G1386,G1387,G1388,G1389,G1390,G1391,G1392,G1393,G1394,G1395,G1396,G1397,G1398,G1399,G1400,G1401,G1402,G1403,G1404,G1405,G1406,G1407,G1408,G1409,G1410,G1411,G1412,G1413,G1414,G1415,G1416,G1417,G1418,G1419,G1420,G1421,G1422,G1423,G1424,G1425,G1426,G1427,G1428,G1429,G1430,G1431,G1432,G1433,G1434,G1435,G1436,G1437,G1438,G1439,G1440,G1441,G1442,G1443,G1444,G1445,G1446,G1447,G1448,G1449,G1450,G1451,G1452,G1453,G1454,G1455,G1456,G1457,G1458,G1459,G1460,G1461,G1462,G1463,G1464,G1465,G1466,G1467,G1468,G1469,G1470,G1471,G1472,G1473,G1474,G1475,G1476,G1477,G1478,G1479,G1480,G1481,G1482,G1483,G1484,G1485,G1486,G1487,G1488,G1489,G1490,G1491,G1492,G1493,G1494,G1495,G1496,G1497,G1498,G1499,G1500,G1501,G1502,G1503,G1504,G1505,G1506,G1507,G1508,G1509,G1510,G1511,G1512,G1513,G1514,G1515,G1516,G1517,G1518,G1519,G1520,G1521,G1522,G1523,G1524,G1525,G1526,G1527,G1528,G1529,G1530,G1531,G1532,G1533,G1534,G1535,G1536,G1537,G1538,G1539,G1540,G1541,G1542,G1543,G1544,G1545,G1546,G1547,G1548,G1549,G1550,G1551,G1552,G1553,G1554,G1555,G1556,G1557,G1558,G1559,G1560,G1561,G1562,G1563,G1564,G1565,G1566,G1567,G1568,G1569,G1570,G1571,G1572,G1573,G1574,G1575,G1576,G1577,G1578,G1579,G1580,G1581,G1582,G1583,G1584,G1585,G1586,G1587,G1588,G1589,G1590,G1591,G1592,G1593,G1594,G1595,G1596,G1597,G1598,G1599,G1600,G1601,G1602,G1603,G1604,G1605,G1606,G1607,G1608,G1609,G1610,G1611,G1612,G1613,G1614,G1615,G1616,G1617,G1618,G1619,G1620,G1621,G1622,G1623,G1624,G1625,G1626,G1627,G1628,G1629,G1630,G1631,G1632,G1633,G1634,G1635,G1636,G1637,G1638,G1639,G1640,G1641,G1642,G1643,G1644,G1645,G1646,G1647,G1648,G1649,G1650,G1651,G1652,G1653,G1654,G1655,G1656,G1657,G1658,G1659,G1660,G1661,G1662,G1663,G1664,G1665,G1666,G1667,G1668,G1669,G1670,G1671,G1672,G1673,G1674,G1675,G1676,G1677,G1678,G1679,G1680,G1681,G1682,G1683,G1684,G1685,G1686,G1687,G1688,G1689,G1690,G1691,G1692,G1693,G1694,G1695,G1696,G1697,G1698,G1699,G1700,G1701,G1702,G1703,G1704,G1705,G1706,G1707,G1708,G1709,G1710,G1711,G1712,G1713,G1714,G1715,G1716,G1717,G1718,G1719,G1720,G1721,G1722,G1723,G1724,G1725,G1726,G1727,G1728,G1729,G1730,G1731,G1732,G1733,G1734,G1735,G1736,G1737,G1738,G1739,G1740,G1741,G1742,G1743,G1744,G1745,G1746,G1747,G1748,G1749,G1750,G1751,G1752,G1753,G1754,G1755,G1756,G1757,G1758,G1759,G1760,G1761,G1762,G1763,G1764,G1765,G1766,G1767,G1768,G1769,G1770,G1771,G1772,G1773,G1774,G1775,G1776,G1777,G1778,G1779,G1780,G1781,G1782,G1783,G1784,G1785,G1786,G1787,G1788,G1789,G1790,G1791,G1792,G1793,G1794,G1795,G1796,G1797,G1798,G1799,G1800,G1801,G1802,G1803,G1804,G1805,G1806,G1807,G1808,G1809,G1810,G1811,G1812,G1813,G1814,G1815,G1816,G1817,G1818,G1819,G1820,G1821,G1822,G1823,G1824,G1825,G1826,G1827,G1828,G1829,G1830,G1831,G1832,G1833,G1834,G1835,G1836,G1837,G1838,G1839,G1840,G1841,G1842,G1843,G1844,G1845,G1846,G1847,G1848,G1849,G1850,G1851,G1852,G1853,G1854,G1855,G1856,G1857,G1858,G1859,G1860,G1861,G1862,G1863,G1864,G1865,G1866,G1867,G1868,G1869,G1870,G1871,G1872,G1873,G1874,G1875,G1876,G1877,G1878,G1879,G1880,G1881,G1882,G1883,G1884,G1885,G1886,G1887,G1888,G1889,G1890,G1891,G1892,G1893,G1894,G1895,G1896,G1897,G1898,G1899,G1900,G1901,G1902,G1903,G1904,G1905,G1906,G1907,G1908,G1909,G1910,G1911,G1912,G1913,G1914,G1915,G1916,G1917,G1918,G1919,G1920,G1921,G1922,G1923,G1924,G1925,G1926,G1927,G1928,G1929,G1930,G1931,G1932,G1933,G1934,G1935,G1936,G1937,G1938,G1939,G1940,G1941,G1942,G1943,G1944,G1945,G1946,G1947,G1948,G1949,G1950,G1951,G1952,G1953,G1954,G1955,G1956,G1957,G1958,G1959,G1960,G1961,G1962,G1963,G1964,G1965,G1966,G1967,G1968,G1969,G1970,G1971,G1972,G1973,G1974,G1975,G1976,G1977,G1978,G1979,G1980,G1981,G1982,G1983,G1984,G1985,G1986,G1987,G1988,G1989,G1990,G1991,G1992,G1993,G1994,G1995,G1996,G1997,G1998,G1999,G2000,G2001,G2002,G2003,G2004,G2005,G2006,G2007,G2008,G2009,G2010,G2011,G2012,G2013,G2014,G2015,G2016,G2017,G2018,G2019,G2020,G2021,G2022,G2023,G2024,G2025,G2026,G2027,G2028,G2029,G2030,G2031,G2032,G2033,G2034,G2035,G2036,G2037,G2038,G2039,G2040,G2041,G2042,G2043,G2044,G2045,G2046,G2047,G2048,G2049,G2050,G2051,G2052,G2053,G2054,G2055,G2056,G2057,G2058,G2059,G2060,G2061,G2062,G2063,G2064,G2065,G2066,G2067,G2068,G2069,G2070,G2071,G2072,G2073,G2074,G2075,G2076,G2077,G2078,G2079,G2080,G2081,G2082,G2083,G2084,G2085,G2086,G2087,G2088,G2089,G2090,G2091,G2092,G2093,G2094,G2095,G2096,G2097,G2098,G2099,G2100,G2101,G2102,G2103,G2104,G2105,G2106,G2107,G2108,G2109,G2110,G2111,G2112,G2113,G2114,G2115,G2116,G2117,G2118,G2119,G2120,G2121,G2122,G2123,G2124,G2125,G2126,G2127,G2128,G2129,G2130,G2131,G2132,G2133,G2134,G2135,G2136,G2137,G2138,G2139,G2140,G2141,G2142,G2143,G2144,G2145,G2146,G2147,G2148,G2149,G2150,G2151,G2152,G2153,G2154,G2155,G2156,G2157,G2158,G2159,G2160,G2161,G2162,G2163,G2164,G2165,G2166,G2167,G2168,G2169,G2170,G2171,G2172,G2173,G2174,G2175,G2176,G2177,G2178,G2179,G2180,G2181,G2182,G2183,G2184,G2185,G2186,G2187,G2188,G2189,G2190,G2191,G2192,G2193,G2194,G2195,G2196,G2197,G2198,G2199,G2200,G2201,G2202,G2203,G2204,G2205,G2206,G2207,G2208,G2209,G2210,G2211,G2212,G2213,G2214,G2215,G2216,G2217,G2218,G2219,G2220,G2221,G2222,G2223,G2224,G2225,G2226,G2227,G2228,G2229,G2230,G2231,G2232,G2233,G2234,G2235,G2236,G2237,G2238,G2239,G2240,G2241,G2242,G2243,G2244,G2245,G2246,G2247,G2248,G2249,G2250,G2251,G2252,G2253,G2254,G2255,G2256,G2257,G2258,G2259,G2260,G2261,G2262,G2263,G2264,G2265,G2266,G2267,G2268,G2269,G2270,G2271,G2272,G2273,G2274,G2275,G2276,G2277,G2278,G2279,G2280,G2281,G2282,G2283,G2284,G2285,G2286,G2287,G2288,G2289,G2290,G2291,G2292,G2293,G2294,G2295,G2296,G2297,G2298,G2299,G2300,G2301,G2302,G2303,G2304,G2305,G2306,G2307,G2308,G2309,G2310,G2311,G2312,G2313,G2314,G2315,G2316,G2317,G2318,G2319,G2320,G2321,G2322,G2323,G2324,G2325,G2326,G2327,G2328,G2329,G2330,G2331,G2332,G2333,G2334,G2335,G2336,G2337,G2338,G2339,G2340,G2341,G2342,G2343,G2344,G2345,G2346,G2347,G2348,G2349,G2350,G2351,G2352,G2353,G2354,G2355,G2356,G2357,G2358,G2359,G2360,G2361,G2362,G2363,G2364,G2365,G2366,G2367,G2368,G2369,G2370,G2371,G2372,G2373,G2374,G2375,G2376,G2377,G2378,G2379,G2380,G2381,G2382,G2383,G2384,G2385,G2386,G2387,G2388,G2389,G2390,G2391,G2392,G2393,G2394,G2395,G2396,G2397,G2398,G2399,G2400,G2401,G2402,G2403,G2404,G2405,G2406,G2407,G2408,G2409,G2410,G2411,G2412,G2413,G2414,G2415,G2416,G2417,G2418,G2419,G2420,G2421,G2422,G2423,G2424,G2425,G2426,G2427,G2428,G2429,G2430,G2431,G2432,G2433,G2434,G2435,G2436,G2437,G2438,G2439,G2440,G2441,G2442,G2443,G2444,G2445,G2446,G2447,G2448,G2449,G2450,G2451,G2452,G2453,G2454,G2455,G2456,G2457,G2458,G2459,G2460,G2461,G2462,G2463,G2464,G2465,G2466,G2467,G2468,G2469,G2470,G2471,G2472,G2473,G2474,G2475,G2476,G2477,G2478,G2479,G2480,G2481,G2482,G2483,G2484,G2485,G2486,G2487,G2488,G2489,G2490,G2491,G2492,G2493,G2494,G2495,G2496,G2497,G2498,G2499,G2500,G2501,G2502,G2503,G2504,G2505,G2506,G2507,G2508,G2509,G2510,G2511,G2512,G2513,G2514,G2515,G2516,G2517,G2518,G2519,G2520,G2521,G2522,G2523,G2524,G2525,G2526,G2527,G2528,G2529,G2530,G2531,G2532,G2533,G2534,G2535,G2536,G2537,G2538,G2539,G2540,G2541,G2542,G2543,G2544,G2545,G2546,G2547,G2548,G2549,G2550,G2551,G2552,G2553,G25
```



```

190 CLS:ID$="ID":GOSUB20
200 PRINT "Give filename":INPUT,B$
210 PRINT "Filename = ";B$
220 PRINT "Correct Y/N"
230 AS=INKEY$:IF AS="Y" THEN 240 ELSE IF AS="N" THEN 200 ELSE 230
240 FD$=AS+E$:ID$="IF":GOSUB20
250 IF IE=0 THEN 260 ELSE PRINT "No such file":PRINT "Directory listing?":PRINT "Y/N"
260 AS=INKEY$:IF AS="Y" THEN 270 ELSE IF AS="N" THEN 200 ELSE 260
270 FOR I=0 TO 29:GETX1,SX,FX,FF$:IF SX=0 THEN IF I=0 THEN PRINT "DIRECTORY EMPTY":GOTO 20
280 CLS:PRINT "Set up 380Z. Call":PRINT "front panel (Ctrl F)"
290 PRINT "option 0, printer 4"
300 INPUT "Give bit rate":BR
310 FOR I=0 TO 6
320 IF BR=B(I) THEN 350
330 NEXT I
340 PRINT "No such bit rate":BR: GOTO 300
350 IF I=1 THEN PRINT "not on the 380Z":GOTO 300
360 PRINT "Bit rate =":BR
370 PRINT "Correct Y/N"
380 AS=INKEY$:IF AS="Y" THEN 390 ELSE IF AS="N" THEN 300 ELSE 380
390 FS="COM0:(" + RIGHT$(STR$(I),1) + "BNIF)"
400 IF I>1 THEN I=I-1:CLS:PRINT "380Z code = ";I
410 PRINT "to return to COM0:Print type R"
420 PRINT "Ready Y/N"
430 AS=INKEY$:IF AS="Y" THEN 440 ELSE IF AS="N" THEN 410 ELSE 430
440 CLS:PRINT "call JWFPI"
450 PRINT "filename.DAT=INP:"
460 PRINT "Ready Y/N"
470 AS=INKEY$:IF AS="Y" THEN 480 ELSE IF AS="N" THEN 440 ELSE 470
480 OPEN "O",E2,F$
490 PRINT E2
500 INPUT E1,G1,G2,AZ,NS,NF,V$
510 FOR I=NS TO NF
520 INPUT E1,I1,R(0,I),R(1,I),R(2,I),R(3,I),R(4,I)
530 NEXT I
540 PRINT E2,"":PRINT E2,USING"\ "V$;PRINT E2,USING"\ "G1;
550 PRINT E2,USING"EEEE"R(0,I);PRINT E2,USING"EEEE"R(1,I);PRINT E2,USING"EEE"AZ;
560 PRINT E2,USING"EE"NS;PRINT E2,USING"EE"NF
570 FOR I=NS TO NF
580 PRINT E2,"":PRINT E2,USING"EEE"R(0,I);PRINT E2,USING"EEEE.EEEE"R(0,I);
590 PRINT E2,USING"EEEE.EEEE"R(1,I);PRINT E2,USING"EEEE.EEEE"R(2,I);
600 PRINT E2,USING"EEEE.EEEE"R(3,I);PRINT E2,USING"EEEE.EEEE"R(4,I)
610 NEXT I
620 PRINT E2,CHR$(8)H1A)
630 IF EOF(1) THEN 640 ELSE 500
640 CLOSE E1:CLOSE E2:PRINT "[C]ontinue or [B]top"
650 AS=INKEY$:IF AS="C" THEN 150 ELSE IF AS="B" THEN ENDELSE 650

```

PROGRAM 3

The purpose of this program is to allow readings from the BGS system to be input, either from a file created by transferring data from the HX-20 or directly entered via the keyboard, and to process the data to produce a Wenner apparent resistivity curve which is then written, unformatted, to a file for interpretation by program 4.

The program assumes that a file containing BGS readings has the qualifier .DAT. If data is being entered via the keyboard then the program creates such a file and writes the unprocessed data to it. The Wenner apparent resistivity curve is written, unformatted, to a file with the same name as the file containing the BGS readings but with the qualifier .INT.

When the program is started the operator is asked whether the data is to be entered from a file. If the answer is yes then the operator is asked for the name of the file and the program checks that the file exists. If the data is to be entered via the keyboard then the operator is asked to give a filename and the program checks that this does not already exist.

If the data is being entered from a file then the program processes all the data on the file sequentially. The only decisions made by the operator are whether the results are listed and a plot of the data is printed. Manually entered data is made in response to prompts very similar to those in program 1. All data entries are echoed by the program on the VDU and must be verified by the operator before they are accepted.

The input data as well as the apparent resistivity curve and the error measures can be listed on the printer. An example of the output from the program is given below.

The subroutine EPSON is a proprietary machine code subroutine. Its function is to dump graphics, in this case plots of the apparent resistivity curve, to the Epson dot-matrix printer.

V.E.S. No. FPS

Grid Ref: BU 61472 89812

Azimuth: 13

INPUT DATA

Electrode Spacing	A	C	D1	D2	B
.5	15.3000	14.7000	11.1000	11.8000	.6840
1.0	7.5500	6.9100	3.9700	3.9700	.3220
2.0	4.8700	4.5400	3.3100	3.4300	.3050
4.0	2.5900	2.4100	1.8800	2.1200	.1700
8.0	1.0900	1.0800	.8500	.8880	.0580
16.0	.8460	.8230	.2750	.2810	.0210

PROCESSING RESULTS

Electrode Spacing	Wenner Resistivity	Observed Error	Offset Error	Lateral Error
.5	39.66	-.0055	.0441	.0837
1.0	35.63	.0158	-.1058	-.1270
1.5	41.60			
2.0	49.61	.0051	-.0291	.0408
3.0	47.40			
4.0	50.27	.0039	.1200	-.0228
6.0	48.97			
8.0	43.68	.0018	.0437	.0825
12.0	34.78			
16.0	27.85	.0058	.0216	0.0000
24.0	22.31			
32.0	18.47			

R.M.S. Observational Error = .0077

R.M.S. Offset Wenner Difference = .0712

R.M.S. Potential Ladder Difference = .0802

The main ~~section~~ of the program handles all data entries and sets up the input and output channels.

```

INTEGER *4 M1,M2
DIMENSION FSP(3),FSP2(3),A(25),RWA(25)
COMMON /A/ RGS(25),RCN(25),PB(25),AA(25),PA(20),PC(20)
DATA YES,RNO,'Y','N'/' ','DAT ',' ','INT '/
DATA ALP,FZ,'0A'/
WRITE(S,600)
600 FORMAT('PROGRAM INPUT '//,
1'This program is used to input resistivity data collected by',//,
2'the Offset-Wenner method to a disc file for interpretation'//)
21 IFLG=1
WRITE(S,908)
908 FORMAT('Do you want to enter data from file? (Y/N)'//)
READ(I,314)ANS
IF (ANS.EQ.RNO) GOTO 921
IF (ANS.NE.YES) GOTO 21
IFLG=0
CALL DISKFL(FSP)
GOTO 32
921 WRITE(S,402)
402 FORMAT('Give the number of the disc drive on which the data file
1is to be created'// 1=A,2=B,3=C,4=D //)
READ(I,403,ERR=921)ID
IF(ID.NE.1.AND.ID.NE.2.AND.ID.NE.3.AND.ID.NE.4) GOTO 921
WRITE(S,318)
318 FORMAT('Enter the primary name of your data file'//)
READ(I,319)FSP(1),FSP(2)
319 FORMAT(2A4)
403 FORMAT(I1)
CALL OPEN(E,FSP,ID)
READ(6,ERR=92)DUMMY
605 WRITE(S,1000)FSP(1),FSP(2),ID
1000 FORMAT('File ',2A4,' already exist on drive ',I2/,
1'Try another name!!')
ENDFILE
GOTO 21
32 WRITE(S,420)
420 FORMAT('Give the number of the disc drive on which the output fi
ile is to be created'// 1=A,2=B,3=C,4=D //)
READ(I,403,ERR=21)ID1
IF(ID1.NE.1.AND.ID1.NE.2.AND.ID1.NE.3.AND.ID1.NE.4) GOTO 32
FSP2(1)=FSP(1)
FSP2(2)=FSP(2)
CALL OPEN(7,FSP2,ID1)
READ(7,ERR=53)DUMMY
650 WRITE(S,1000)FSP2(1),FSP2(2),ID1
ENDFILE
GOTO 21
C C C
ENTER THE DETAILS AND RESISTIVITY READINGS TO DISK FILE
55 IF (IFLG.GT.965,906,965)
906 READ(6,3,END=82) AIP1,AIP2,CRSJ,M1,M2,NZ,NS,NF
N=N+NF
STAT=N*.5
DO 907 I=1,N
AA(I)=O*((I-1)*STAT)
READ(6,108) II,PA(I),PG(I),RGS(I),RCN(I),PB(I)
907 CONTINUE
GOTO 952
965 WRITE(S,301)
301 FORMAT(X,'Give the V.E.S. number - (up to 8 characters)'//)
READ(I,302) AIP1,AIP2
302 FORMAT(2A4)
WRITE(S,490) AIP1,AIP2
490 FORMAT('You have entered the V.E.S. number as ',2A4)
28 WRITE(S,313)
READ(I,314)ANS
IF (ANS.EQ.RNO) GOTO 965
IF (ANS.NE.YES) GOTO 28
50 WRITE(S,309)
309 FORMAT(X,'Give the grid letters of the sounding grid reference'//)

```

```

      READ(1,304) GPSJ
22  WRITE(5,405)
      READ(1,406,ERR=22) M1
23  WRITE(5,407)
      READ(1,406,ERR=23) M2
406  FORMAT(I5)
405  FORMAT(' Give the easting of the soundings 5-figure grid reference
1//)
407  FORMAT(' Give the northing of the soundings 5-figure grid reference
1e//)
      WRITE(5,408) CRSJ,M1,M2
408  FORMAT(' The grid reference you have entered is ',A2,217)
29  WRITE(5,313)
      READ(1,314) ANS
      IF (ANS.EQ.RNO) GOTO 60
      IF (ANS.NE.YES) GOTO 29
51  WRITE(5,305)
304  FORMAT(A2)
305  FORMAT(IX, ' Give the azimuth of the sounding (up to three digits)'/
1)
      READ(1,306,ERR=61) NZ
306  FORMAT(I3)
50  WRITE(5,307)
307  FORMAT(IX, ' Give the first spacing used (up to two digits)'/)
      READ(1,308,ERR=50) NS
308  FORMAT(I2)
      WRITE(5,320)
320  FORMAT(IX, ' Give the last spacing used (up to two digits)'/)
      READ(1,308,ERR=50) NF
      WRITE(5,500) NZ,NS,NF
500  FORMAT(' You have entered the azimuth as 'I4/' the first spa
1cings as 'I3/' the last spacing as 'I3//)
40  WRITE(5,313)
      READ(1,314) ANS
      IF (ANS.EQ.RNO) GOTO 61
      IF (ANS.NE.YES) GOTO 40
      N=NF-NS+1
31  WRITE(5,312) N
3:2  FORMAT(' Give ',I2,' sets of readings (up to 10 digits including
1 a decimal point)'/)
      DO 5 I=1,N
52  WRITE(5,410) I
      READ(1,201,ERR=62) PA(I)
24  WRITE(5,411) I
      READ(1,201,ERR=24) PG(I)
25  WRITE(5,412) I
      READ(1,201,ERR=25) RCS(I)
26  WRITE(5,413) I
      READ(1,201,ERR=26) RGN(I)
27  WRITE(5,414) I
      READ(1,201,ERR=27) PB(I)
410  FORMAT(' Give the A reading for spacing ',I2//)
411  FORMAT(' Give the C reading for spacing ',I2//)
412  FORMAT(' Give the D1 reading for spacing ',I2//)
413  FORMAT(' Give the D2 reading for spacing ',I2//)
414  FORMAT(' Give the B reading for spacing ',I2//)
480  FORMAT(5X, 'WARNING: The observed error is greater than 1X//,14X, 'T
1here may be an incorrect input data value'//)
      EE=(PA(I)-PB(I)-PG(I))/PA(I)
      IF (ABS(EE)-0.01) 90,90,91
91  WRITE(5,480)
90  WRITE(5,415) I, PA(I),PG(I),RCS(I),RGN(I),PB(I)
415  FORMAT(' The readings for electrode spacing ',I2,' you have entered
1 are'/5F12.4)
41  WRITE(5,313)
      READ(1,314) ANS
      IF (ANS.EQ.RNO) GOTO 62
      IF (ANS.NE.YES) GOTO 41
5  CONTINUE
313  FORMAT(IX, 'Correct? Y or N ?//)
314  FORMAT(A1)
201  FORMAT(F10.0)

52  WRITE (5,93) AIP1,AIP2,CRSJ,M1,M2,NZ,NS,NF
93  FORMAT(IX,2A4,A2,215,I3,212)
      STAT=0.5*NS

```



```

DO 54 I=1,N
AA(I)=2.0*(I-1)*STAT
WRITE(6,108) I,PA(I),PG(I),RGS(I),RCN(I),PB(I)
108 FORMAT(14,5F9.4)
54 CONTINUE
952 CALL BGS(AIP1,AIP2,GRSJ,M1,M2,NZ,N,M,A,RWA)
CALL LOG3
CALL RESPLT(A,RWA,M,0.0,0.0)
WRITE(5,555)
555 FORMAT(1H1)
WRITE(7)AIP1,AIP2,GRSJ,M1,M2,NZ,M
DO 70 I=1,M
WRITE(7)A(I),RWA(I)
70 CONTINUE
901 WRITE(5,900)
900 FORMAT(' Do you want the plot printed? (Y/N)')
READ(1,314)ANS
IF (ANS.EQ.'N') GOTO 42
IF (ANS.NE.'Y') GOTO 901
WRITE(2,902)
902 FORMAT(1X,' PLOTTED RESULTS')
CALL EPSON(0,2)
42 IF (IFLG) 942,906,942
942 WRITE(5,900)
800 FORMAT(' Do you wish to enter more field data? (Y or N)')
READ(1,314)ANS
IF (ANS.EQ.'N') GOTO 82
IF (ANS.NE.'Y') GOTO 42
GOTO 965
82 WRITE(5,83)FSP(1),FSP(2)
83 FORMAT(' Your field data has been written to file ',2A4//)
ENDFILE 6
ENDFILE 7
CALL RESOL(0,2)
STOP
END

```

Subroutine BGS

This subroutine processes the input readings from the BGS offset Wenner system to produce a Wenner apparent resistivity curve. The variables used in the call are:

AIP1,AIP2 - the sounding reference number
 GRSJ - the UK grid reference letters
 M1 - grid reference easting
 M2 - grid reference northing
 NZ - azimuth of the array
 N - the number of electrode spacings used
 M - the number of electrode separations on the processed curve
 A - output vector containing the electrode separations
 RWA - output vector containing the Wenner apparent resistivities

The BGS offset readings are entered to the subroutine using vectors through the common block A. The vectors are:

RGS - D1 readings
 RCN - D2 readings
 PB - B readings
 PA - A readings
 PG - C readings
 AA - electrode separations

```

SUBROUTINE BGS(AIP1,AIP2,GRSJ,M1,M2,NZ,N,M,A,RWA)
INTEGER *4 M1,M2
DIMENSION RA(25),RB(25),RC(25),E(25),RWB(20),RWC(20),
1RC1(20),RC2(20),EL(20),H(20),A(25),RWA(25)
COMMON /A/ RGS(25),RCN(25),PB(25),AA(25),PA(20),PG(20)
DATA YES,RNO/'Y','N'/
DATA CR/'OD'/
STAT=AA(1)
IFLG=0
PI=4.0*ATAN(1.0)
A(1)=STAT
DO 20 I=1,10
J=I+2
K=J+1
A(J)=2.0*I*STAT
A(K)=3.0*STAT*2.0*(I-1)
20 CONTINUE
WRITE(5,60)
60 FORMAT(' Do you wish to print processed results? (Y or N)')
READ(1,61)ANS
61 FORMAT(A1)
IF (ANS.EQ.'N') GOTO 70
IF (ANS.NE.'Y') GOTO 20
93 FORMAT(' V.E.S. NO. 1X,2A4,15X, 101D R1, 1X,2E, 1X,1D,1X,1D,4X
1,9HAzimuth: ,13//')
104 FORMAT(22X,' INPUT DATA'//,4X,'Electrode',5X,'Spacing',5X,1HA,9X,
11HC,9X,2HD1,8X,2HD2,8X,1HB//)
WRITE(2,77)
77 FORMAT(1H1)
WRITE(2,93) AIP1,AIP2,GRSJ,M1,M2,NZ
WRITE(2,104)
DO 54 I=1,N
WRITE(2,108) AA(I),PA(I),PG(I),RGS(I),RCN(I),PB(I)
54 CONTINUE
108 FORMAT(4X,F6.1,5F10.4)
109 FORMAT(/4X,9HElectrode,4X,6HWhenor,4X,8H0bserved,2X,6H0ffset,
13X,7HLateral/5X,7HSpacing,2X,11HResistivity,3(4X,5HError)//)
113 FORMAT(4X,F6.1,3X,F9.2,3X,3F9.4)
114 FORMAT(4X,F6.1,3X,F9.2)
70 EEE=0.0
HHH=0.0
DO 32 I=1,N
E(I)=PA(I)-PB(I)-PG(I)
EE=ABS(PA(I))+ABS(PB(I))+ABS(PG(I))
RA(I)=PA(I)-E(I)*PA(I)/EE
RB(I)=PB(I)+E(I)*PB(I)/EE
RC(I)=PG(I)+E(I)*PG(I)/EE
RC1(I)=(RCN(I)+RGS(I))/2.0
H(I)=(RCN(I)-RGS(I))/RC1(I)
RC2(I)=RC(I)-RC1(I)
E(I)=E(I)/RA(I)
EEE=EEE+E(I)*E(I)
HHH=HHH+H(I)*H(I)
32 CONTINUE
EEE=SQRT(EEE/FLOAT(N))
HHH=SQRT(HHH/FLOAT(N))
RWA(1)=RC1(1)
K=N-1
DO 3 I=2,K
I1=I+2-1
I2=I+1
I3=I-1
I4=I-2
RWA(I2)=RC1(I)
RWA(I1)=RC1(I)/2.0+RB(I)-RB(I1)+RC1(I1)/2.0
3 CONTINUE
I1=N*2-1
I2=N*2
NN=N*2-2
RWA(NN)=RC1(N)
RWA(I1)=RC1(N)/2.0+PB(N)-RB(K)+RC2(N)
PWA(I1)=PG2(N)*2.0
ELL=0.0
DO 4 I=1,K
I1=I+1
EL(I)=(RC2(I)-RC1(I1)/2.0)/RC1(I1)
ELL=ELL+EL(I)*EL(I)

```

```

4 CONTINUE
ELL=SQRT(ELL/FLOAT(K))
EL(N)=0.0
M=N*2
DO 33 I=1,M
RWA(I)=2.0*PI*A(I)*RWA(I)
33 CONTINUE
IF(ANS.EQ.RNO)GOTO 93
WRITE(2,115)
115 FORMAT(///17X, 'PROCESSING RESULTS')
WRITE(2,109)
WRITE(2,113) A(1), RWA(1), E(1), H(1), EL(1)
DO 6 I=2,N
II=I*2-2
IJ=I*2-1
WRITE(2,113) A(IJ), PWA(IJ), E(I), H(I), EL(I)
WRITE(2,114) A(IJ), PWA(IJ)
6 CONTINUE
480 FORMAT(5X, 'WARNING: The observed error for spacing 1.2' is greater
1 than 1% //, 14X, 'There may be an incorrect input data value //')
WRITE(2,114) A(M), RWA(M)
WRITE(2,110) EEE
110 FORMAT(//13X, 29HR.M.S. Observational Error = ,F5.4)
WRITE(2,111) HHH
111 FORMAT(8X, 34HR.M.S. Offset Wenner Difference = ,F5.4)
WRITE(2,112) ELL
112 FORMAT(5X, 37HR.M.S. Potential Ladder Difference = ,F5.4/)
95 DO 80 I=1,N
IF(ABS(E(I))-0.01) 80,80,85
85 WRITE(5,480)I
80 CONTINUE
DO 4539 I=1,M
IF(RWA(I).LE.0.0)GOTO 4540
4539 CONTINUE
GOTO 9999
4540 M=-1
WRITE(5,9991)
IF(ANS.EQ.YES)WRITE(2,9991)
9991 FORMAT(// 'NEGATIVE RESISTIVITIES CANNOT INTERPET//')
9999 RETURN
END

```

Subroutine LG2LG3

This subroutine produces a log/log grid on the 380Z's VDU in high resolution graphics mode. The grid is two cycles vertically by three cycles horizontally and is drawn in red:

```

SUBROUTINE LG2LG3
BYTE CLT(16)
CALL RESOL(0,2)
CLT(1)=MIX(0,0,0)
CLT(2)=MIX(7,0,0)
CLT(3)=MIX(0,7,0)
CLT(4)=MIX(0,0,2)
CALL COLOUR(CLT)
CALL PLOT(79,190,1)
CALL LINE(79,31,1)
CALL LINE(318,31,1)
DO 20 I=1,3
DO 21 J=1,10
IX1=ALOG10(J*10.0*(I-1))*80+79
CALL PLOT(IX1,31,1)
CALL LINE(IX1,190,1)
21 CONTINUE
20 CONTINUE

```

```

DO 22 I=1,2
DO 23 J=1,10
IX1=ALOG10(J*10.0*(I-1))*80+31
CALL PLOT(79,IX1,1)
CALL LINE(318,IX1,1)
23 CONTINUE
22 CONTINUE
RETURN
END

```

Subroutine RESPLT

This subroutine plots the data points of an apparent resistivity curve on the 380Z's VDU using the high resolution graphics. The points are plotted in blue. The variables used by the subroutine are:

X - vector containing the electrode separations
Y - vector containing the apparent resistivities
M - the number of data points

Values returned by the subroutine are:

XST - the value of the leftmost log cycle
AM - the logarithm of the mean apparent resistivity value

```

SUBROUTINE RESPLT(X,Y,M,XST,AM)
DIMENSION X(M),Y(M)
IF (XST) 32,31,32
31 IF (AM) 32,30,32
30 AM=ALOG10(Y(1))
DO 20 I=2,M
AM=AM+ALOG10(Y(I))
20 CONTINUE
AM=IFIX(AM/FLOAT(M))
XST=ALOG10(X(1))
IF (XST) 21,22,22
21 XST=XST-1.0
22 XST=AIN(XST)
32 DO 23 I=1,M
IX=80*(ALOG10(X(I))-XST)+79
IY=80*(ALOG10(Y(I))-AM)
26 IF (IY) 24,25,25
24 IY=IY+160
25 IF (IY-160) 27,27,28
28 IY=IY-160
27 IY=IY+31
CALL PLOT(IX,IY,3)
23 CONTINUE
RETURN
END

```

Subroutine DISKFL

The purpose of this subroutine is to open a disc file on a specified drive. Questions are displayed on the VDU and answers given by the operator through the keyboard are used to control the file that is opened. There is only one vector returned, FSP, which contains the name of the file.

```

SUBROUTINE DISKFL(FSP)
  DIMENSION FSP(3)
  DATA YES,RNO/'Y',N/'N'
21 WRITE(5,402)
402 FORMAT(' Give the number of the disc drive on which the data file
1 is located // 1=A,2=B,3=C,4=D //')
  READ(1,403)ID
  IF(ID.NE.1.AND.ID.NE.2.AND.ID.NE.3.AND.ID.NE.4) GOTO 21
403 FORMAT(11)
66 WRITE(5,318)
318 FORMAT(1X,'Enter the primary name of the data file //')
  READ(1,319)FSP(1),FSP(2)
319 FORMAT(2A4)
  CALL OPEN(6,FSP,1D)
  READ(6,ERR=33,END=33)DUMMY
  GOTO 99
23 WRITE(5,44)FSP(1),FSP(2),FSP(3),ID
44 FORMAT(' File ',2A4,' ',A4,' is empty or does not exist on drive ',
111// ' Do you wish to continue? (Y or N) //')
  READ(1,55)ANS
55 FORMAT(A1)
  IF(ANS.EQ.RNO)STOP
  IF(ANS.NE.YES)GOTO 33
  GOTO 21
99 REWIND 6
  RETURN
END

```

PROGRAM 4

This program is based on the work of Koefoed (1979a), but the algorithm that carries out the interpretation has been modified to improve the speed as recommended by Koefoed (1979b). In addition, routines have been added that allow the program to handle data with irregular electrode spacings and the input and output routines added to make the system user-friendly. Koefoed's program is designed for the Schlumberger array but this program is designed to handle both Wenner and Schlumberger array data. Ideally this should have been done by the application of a linear filter to convert Wenner array data to Schlumberger array data. However, the limit of the computer's memory made this impractical. Instead Wenner array data is merely shifted in the x-axis by the square root of 2. This approximation only produces unsatisfactory results if very large differences in the resistivity of adjacent layers are encountered.

The program uses an iterative interpretation method based on the method of steepest descents. This method, when applied to resistivity sounding interpretation, yields a fast improvement in the layer parameters. The field data are compared with data calculated from a layered model and if the agreement between the two data sets is unsatisfactory, the parameters of the layered model are adjusted in the direction of the "steepest descent" of the difference between the two and the procedure repeated. Unless a starting model is used which is a rough approximation to the final solution, the iterations either fail to converge or do so slowly. However, with experience the operator can provide an adequate starting model merely by examining the field data displayed on the VDU by the program.

The apparent resistivity curve is calculated from the layered model by firstly taking the transform and then applying a linear filter, as described by Ghosh

(1971). This method allows the calculations to be made rapidly with only a small error, generally less than 1%, due to the approximations involved.

The program is designed to be fully interactive and leads the operator through the necessary procedures. All data entries from the keyboard, or decisions for output requirements are made by the operator in response to prompts by the program. In order to minimise operator errors, all data entries are echoed by the program and must be verified by the operator before the program will proceed.

The first section of the program concerns data entry. Data is entered from disc files in an unformatted state. When the program is started the operator is asked to give the name of the data file and which disc drive it is located on. The program checks whether the file exists before proceeding. The data can be interpreted sequentially, i.e. the soundings are dealt with one after another, or soundings specified by the operator.

After a sounding has been selected for interpretation, the field curve is plotted on the VDU and the operator is offered the chance of editing the data. This can consist of deleting, adding or altering points. Experience has shown that editing usually consists of deleting obviously spurious data points.

The operator is asked to input the number of layers for the starting model and then the thicknesses of the layers, followed by their resistivities. The program cannot change the number of layers, only the layer parameter value. In addition the program provides the facility for assigning fixed values for the layer resistivities and/or the depth to the boundary between two layers by adding 100000 to the value. This allows the operator to incorporate knowledge of the local geology, such as the depth to a particular boundary derived from a borehole close to the site of the sounding.

The interpretation algorithm is then called by the program. The iterations are carried out five at a time, after which the operator is presented with the option of terminating the iteration. Otherwise the iteration is terminated either when 30 have been completed or when the difference between the field data and the values calculated from the layered model falls below a level entered earlier by the operator. Iteration also terminates if the adjustments made to the layered model fall below a level set within the program.

On completing the iterations the calculated curve and the field data are displayed on the VDU. The operator is given the option of printing the final layered model and the plot of the curves, before trying another starting model for that sounding, starting to interpret another set of field data, or stopping. An example of the results of an interpretation is shown below.

The data upon which the interpretation is based is listed, as is the final layered model. The R.M.S. relative error is the average difference between the field and calculated apparent resistivity curves and thus is a measure of the quality of the interpretation. The value that can be attained is dependant on the quality of the field data, but, the smaller the value, the better the interpretation. The number of trials is the number of iterations carried out to produce the final model.

V.E.B. No. FP9

FIELD CURVE DATA			INTERPRETED MODEL		
Electrode Separation	Apparent Resistivity		Depth	Rho	Reflection Coeffts.
1 .5	35.66	1.42	1.42	34.1	.3779
2 1.0	35.63				
3 1.5	41.60				
4 2.0	49.61				
5 3.0	47.40	3.40	3.40	75.6	.5338
6 4.0	50.27				
7 6.0	48.97				
8 8.0	43.68				
9 12.0	34.78	25.19	25.19	23.0	.6264
10 16.0	27.95				
11 24.0	22.31				
12 32.0	18.47				
				5.3	

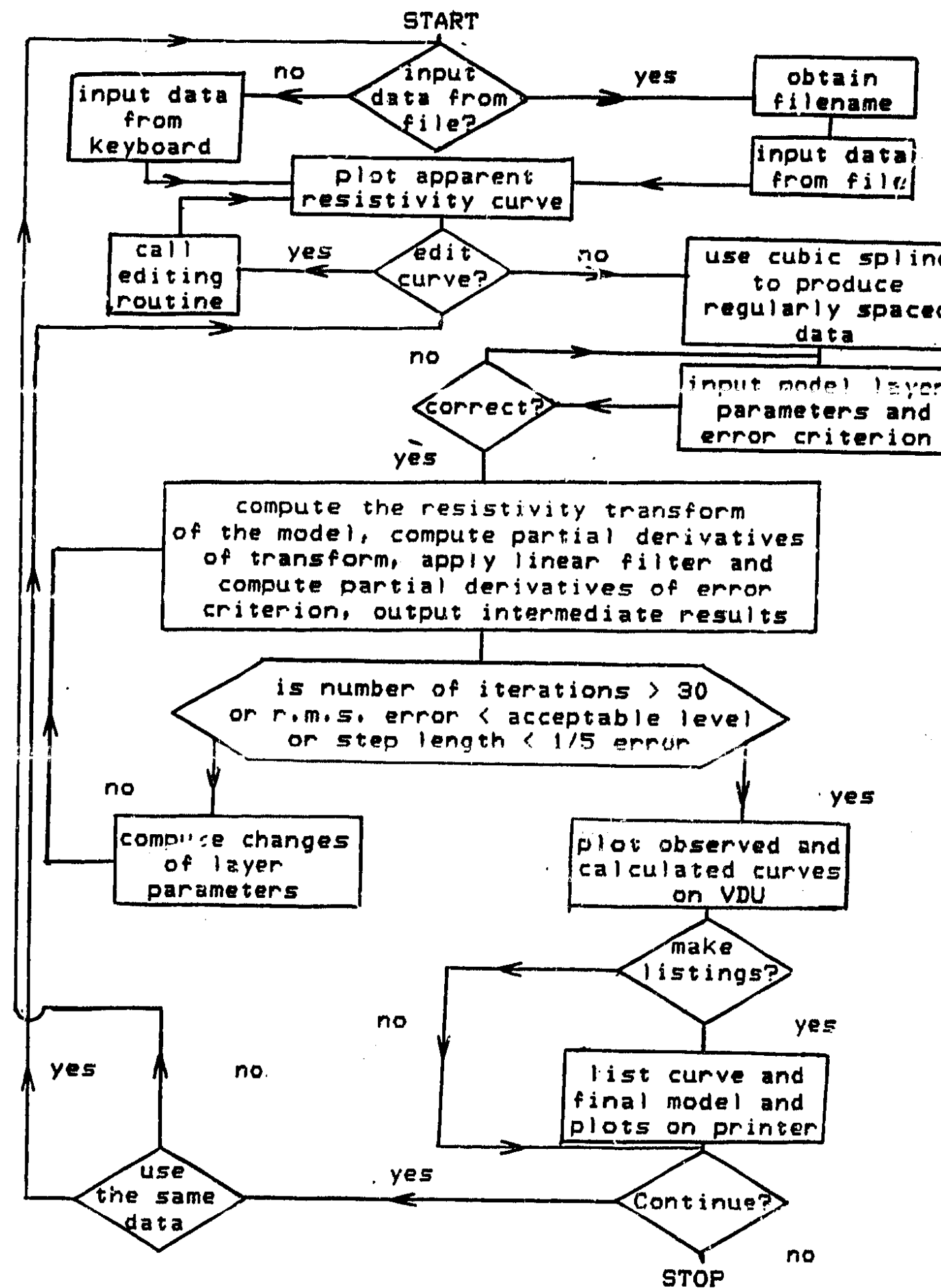
R.M.S. Relative error = .0313
 Maximum rel. error = .0801 at sample 2
 Number of trials was 14

The main segment of the program handles setting up the input files and selecting the sounding to be interpreted before handing control to the interpretation algorithm.

```

LOGICAL FLG
INTEGER *4 M1,M2
DIMENSION A(25),RWA(25)
DATA YES,RNO,'Y','N'
CALL RESOL(0,2)
25 WRITE(5,500)
500 FORMAT(' Which electrode array? (W)enner or (S)chlumberger')
READ(1,10,ERR=25) WSA
FLG=.TRUE.
700 CALL DISKFL
30 WRITE(5,2)
2 FORMAT(' Do you wish to start interpreting data from the',
1 ' beginning of the data file? (Y or N)')
READ(1,10)ANS
10 FORMAT(A1)
IF(ANS.EQ.RNO)GOTO 130
IF(ANS.NE.YES)GOTO 30
GOTO 40
220 FLG=.TRUE.
WRITE(5,210)
210 FORMAT(' Do you want to interpret the next sounding? (Y or N)')
READ(1,10)ANS
IF(ANS.EQ.RNO)GOTO 130
IF(ANS.NE.YES)GOTO 220
GOTO 40
130 FLG=.FALSE.
40 CALL SELECT(AIP1,AIP2,GRB,M1,M2,NZ,M,FLG,A,RWA)
IF (N) 90,70,90
70 WRITE(5,100)
100 FORMAT(' Do you wish to interpret any other soundings? (Y or N)')

```



```

      READ(1,10)ANS
      IF(ANS.EQ.RNO)GOTO 99
      IF(ANS.NE.YES)GOTO 70
      IF(N.EQ.0)GOTO 750
      GOTO 220
750 WRITE(5,760)
760 FORMAT(' Do you wish to interpret soundings from a different file',
      1 ' (Y or N)')
      READ(1,10)ANS
      IF(ANS.EQ.RNO)GOTO 220
      IF(ANS.NE.YES)GOTO 750
      ENDFILE 6
      GOTO 700
90 CALL VESINT(AIP1,AIP2,GRSJ,M1,M2,M,A,RWA,WSA)
      GOTO 70
99 STOP
      END

```

Subroutine DISKFL

This subroutine opens a disc file on stream 6. The drive number and file name are given from the keyboard by the operator. If the file does not exist the operator is prompted to supply another name.

```

      SUBROUTINE DISKFL
      DIMENSION FSP(3)
      DATA FSP/' ',' ','INT'
      DATA YES,RNO/'Y ','N'
21 WRITE(5,402)
402 FORMAT(' Give the number of the disc drive on which the data file
      1 is located'/' 1=A,2=B,3=C,4=D')
      READ(1,403)ID
      IF(ID.NE.1.AND.ID.NE.2.AND.ID.NE.3.AND.ID.NE.4) GOTO 21
403 FORMAT(11)
66 WRITE(5,318)
318 FORMAT(1X,'Enter the primary name of the data file')
      READ(1,319)FSP(1),FSP(2)
319 FORMAT(2A4)
      CALL OPEN(6,FSP,ID)
      READ(6,ERR=33,END=33)DUMMY
      GOTO 99
33 WRITE(5,44)FSP(1),FSP(2),FSP(3),ID
44 FORMAT(' File ',2A4,', ',A4,' is empty or does not exist on drive ',
      111' Do you wish to continue? (Y or N)')
      READ(1,55)ANS
55 FORMAT(A1)
      IF(ANS.EQ.RNO)STOP
      IF(ANS.NE.YES)GOTO 33
      GOTO 21
99 REWIND 6
      RETURN
      END

```

Subroutine SELECT

This subroutine selects resistivity sounding data from a sequential file. If the selection fails all the V.E.S. numbers on the file can be listed. The inputs are provided directly from the keyboard. The output variables are:

AIP1,AIP2 - the V.E.S. reference number
 GRSJ - the UK grid reference letters
 M1 - the grid reference easting
 M2 - the grid reference northing
 NZ - the azimuth of the array
 N - the number of points on the apparent resistivity curve
 FLG - if zero the next V.E.S. is taken
 A - vector containing the electrode separations
 RWA - vector containing the apparent resistivity curve

```

      SUBROUTINE SELECT(AIP1,AIP2,GRSJ,M1,M2,NZ,N,FLG,A,RWA)
      LOGICAL FLG
      INTEGER *4 M1,M2
      DIMENSION A(25),RWA(25)
      DATA BL,YES,RNO/' ','Y ','N'
      IF(FLG) GOTO 3
50 REWIND 6
9 WRITE(5,1)
1 FORMAT(' Enter V.E.S. number you require')
      READ(1,4)RAIP1,RAIP2
4 FORMAT(2A4)
      IF(RAIP1.EQ.BL.AND.RAIP2.EQ.BL)GOTO 9
3 READ(6,END=11,ERR=77)AIP1,AIP2,GRSJ,M1,M2,NZ,N
      DO 25 I=1,N
      READ(6,ERR=77) A(I),RWA(I)
25 CONTINUE
600 IF(FLG) GOTO 12
      IF(AIP1.EQ.RAIP1.AND.AIP2.EQ.RAIP2) GOTO 12
      GOTO 3
77 WRITE(5,78)
78 FORMAT(' READ ERROR: May be corrupt data file')
      STOP
11 IF(FLG)GOTO 99
      WRITE(5,13)RAIP1,RAIP2
13 FORMAT(' No such V.E.S. number as ',2A4/,
      1 ' Do you wish to list v.e.s. numbers on the file? (Y or N)')
      READ(1,14,ERR=11)ANS
14 FORMAT(A1)
      IF(ANS.EQ.RNO)GOTO 20
      IF(ANS.NE.YES)GOTO 11
      REWIND 6
34 ICOUNT=0
31 READ(6,END=20)AIP1,AIP2
      IF(AIP1.EQ.BL.AND.AIP2.EQ.BL)GOTO 31
      WRITE(5,6)AIP1,AIP2
6 FORMAT(1H,2A4)
      ICOUNT=ICOUNT+1
      IF(ICOUNT.EQ.15)GOTO 33
      GOTO 31
33 WRITE(5,55)
55 FORMAT(' Continue listing? (Y or N)')
      READ(1,51)ANS
51 FORMAT(A1)
      IF(ANS.EQ.RNO)GOTO 20
      IF(ANS.NE.YES)GOTO 33
      GOTO 34
20 WRITE(5,32)
32 FORMAT(' Do you wish to continue selection (Y or N)')
      READ(1,14)ANS
      IF(ANS.EQ.RNO)STOP
      IF(ANS.NE.YES)GOTO 20
      GOTO 50
99 N=0
      WRITE(5,333)
333 FORMAT(' End of file. no more soundings')
12 RETURN
      END

```

Subroutine VESINT

This subroutine is the interpretation algorithm. It also calls the editing and output subroutines. The input variables are:

AIPI, AIP2 - the V.E.S. reference number
 GRSJ - the UK grid reference letters
 M1 - the grid reference easting
 M2 - the grid reference northing
 JZZ - the number of points on the apparent resistivity curve
 OAB - vector containing the electrode separations
 OAR - vector containing the apparent resistivity curve
 WSA - variable denoting the type of array

```

SUBROUTINE VESINT(AIP1, AIP2, GRSJ, M1, M2, JZZ, OAB, OAR, WSA)
  INTEGER*4 M1, M2, MJ
  DIMENSION RF(25), P(19), PL(19), C(19), GL(19), T(9), U(19), D(4, 19)
  1, AB(25), RHO(10), DEPTH(10), RD(10), REFCO(10), W(75), AR(25), C(25)
  2, RMM(25), ZZB(25), OAB(25), OAR(25)
  DATA YES, RNO, CA, WAR, SAR, Y, N, C, W, S, /
  F=EXP(ALOG(10.)/6.)
  RA=.75
  SQT=.SQRT(2.0)
  RB=.6
  XST=0.0
  AM=0.0
  IFLG=0
  JFLG=0
500 WRITE(5, 2000)
2000 FORMAT(1H1)
  CALL LC2LC3
  CALL RESPLT(OAB, OAR, JZZ, XST, AM)
C
  INUMB=0
  WRITE(5, 1007) AIP1, AIP2
  DO 501 J=1, JZZ
    WRITE(5, 1006) J, OAB(J), OAR(J)
    AB(J)=OAB(J)
    IF (WSA-SAR) 567, 568, 567
567 AB(J)=AB(J)*SQT2
568 AB(J)=ALOG10(AB(J))
    AR(J)=ALOG10(OAR(J))
501 CONTINUE
    WRITE(5, 1010)
    READ(1, 1002) ANS
    IF (ANS.EQ.RNO) GOTO 505
    IF (ANS.NE.YES) GOTO 501
    GOTO 6
505 IF (IFLG) 168, 506, 168
    6 CALL EDITXY(OAB, OAR, JZZ)
    IFLG=0
    GOTO 500
506 XA=10.0**AB(1)
    WRITE(5, 2000)
    JZ=JZZ
    JZ3=3*JZ
    CALL CUSP1(JZ, JZ3, AB, AR, C, W)
    IF (W(1)) 166, 167, 166
167 IE=-1
    IFLG=1
    DO 160 I=1, 25
      ZB=AB(1)+FLOAT(I-1)/6.0
      ZZB(I)=OAB(1)
      IF (I-1) 600, 161, 600
600 ZZB(I)=ZZB(I)+10.0*(FLOAT(I-1)/6.0)
161 RF(I)=CUSP2(IE, JZ, AB, AR, C, ZB)
      IE=1
      RF(I)=10.0**RF(I)
      IF (ZB-AB(JZ)) 160, 160, 162
160 CONTINUE

```

```

162 JZ=J-1
168 WRITE(5, 132)
  READ(1, 1001, END=168) E
  IF (E.GT.1.0.OR.E.LT.0.0) GOTO 168
C
C   WRITE OUT LAST MODEL PARAMETERS
C
  IF (JFLG.EQ.0) GOTO 633
  WRITE(5, 1234)
  WRITE(5, 1237) (RD(IK1), RHO(IK1), IK1=1, IK)
  WRITE(5, 1238) RHO(IL)
C
633 WRITE(5, 138)
  READ(1, 1000, ERR=633) IL
  IF (IL.LE.1.OR.IL.GT.10) GOTO 633
  WRITE(5, 3005)
3005 FORMAT(/' If you do not want a layer parameter to be adjusted, add
1 100000 to its value'/' When entering a layer parameter use up to
210 digits including a decimal point'/'
  IK=IL-1
  IT=IK+IL
  8 DO 9 I=1, IK
634 WRITE(5, 139) I
    READ(1, 1003, ERR=634) P(I)
    IF (P(I).LE.0.0) GOTO 634
    9 CONTINUE
    DO 10 I=IL, IT
      II=I-IL+1
635 WRITE(5, 140) II
      READ(1, 1003, ERR=635) P(I)
      IF (P(I).LE.0.0) GOTO 635
    10 CONTINUE
    WRITE(5, 3010)
3010 FORMAT(/'Layer', 4X, 'Thickness', 2X, 'Resistivity')
    DO 96 I=1, IK
      II=IL+IK
      WRITE(5, 141) I, P(I), P(II)
96 CONTINUE
      WRITE(5, 142) IL, P(II)
3002 FORMAT('Layer', 17X, F10.2)
625 WRITE(5, 137)
  READ(1, 1002) ANS
  IF (ANS.EQ.RNO) GOTO 8
  IF (ANS.EQ.CA) GOTO 168
  IF (ANS.NE.YES) GOTO 625
11 WRITE(5, 2000)
  QV=99.0
  ST=0.0
  IC=0.0
  IM=0.0
  DO 13 I=1, IT
    U(I)=1.0
13 CONTINUE
C
C   COMPUTE THE RESISTIVITY TRANSFORM FOR THE CURRENT LAYER
C   MODEL
C   T(I)
16 XR=XA
  J=1
  Q=0.0
  BM=0.0
  DO 17 I=1, IT
    G(I)=0.0
17 CONTINUE
18 X=0.0105*XR
  LB=1
  DO 30 L=1, 9
    X=X*F
19 B=P(IT)
    IF (100000.0-B) 20, 20, 21
20 B=B-100000.0
21 BC=B
    DO 29 K=1, IK
      ILK=IL-K
      ITK=IT-K
      DW=P(ILK)

```



```

RW=P/(TK)
IF (100000.0-DW) 22,22,23
22 DW=DW-100000.0
23 IF (100000.0-RW) 24,24,25
24 RW=RW-100000.0
25 TH=TANH(DW/X)
26 B=(B+TH*RW)/(1.+TH*B/RW)
30 T=L-2

```

```

      COMPUTE THE PARTIAL DERIVATIVES OF THE SAMPLE VALUES OF THE
      RESISTIVITY TRANSFORM WITH RESPECT TO THE LAYER PARAMETERS
      D(I,K)

```

```

31 X=.1*X
  L=4
  IF (LB-1) 32,32,33
32 X=.1*X
  DO 41 L=LB, 4
33 D(L,IT)=1.0
  B=PC
  DO 40 K=1, IK
    IW=IT-K
    IX=IW+1
    DW=P(IX)
    RW=P(IW)
    IF (100000.0-DW) 34,34,35
34 DW=DW-100000.0
35 IF (100000.0-RW) 36,36,37
36 RW=RW-100000.0
37 TH=TANH(DW/X)
  BA=1.+B*TH/RW
  PA=(1.-TH*TH)/(BA*BA)
  D(L,1)=(RW-B*B/RW)/X
  D(L,IW)=TH*(1.+B*B/(RW*RW)+2.*TH*B/RW)/(BA*BA)
  DO 38 IZ=1, IK
    D(L,IZ)=PA*D(L,IZ)
38 CONTINUE
  DO 39 IZ=1Y, IT
    D(L,IZ)=PA*D(L,IZ)
39 CONTINUE
  B=(B+TH*RW)/(1.+TH*B/RW)
40 CONTINUE
  X=X*F*F
41 CONTINUE

```

```

      APPLY THE LINEAR FILTERS AND COMPUTE THE PARTIAL DERIVATIVES
      OF THE ERROR CRITERION      G(I)

```

```

43 RM=C.0148*T(1)-0.0814*T(2)+0.4018*T(3)-1.5716*T(4)+1.972*T(5)+0.18
154*T(6)+0.1064*T(7)-0.0499*T(8)+0.0225*T(9)
  RMN(J)=RM
  DO 46 L=1, 3
    T(L)=T(L+1)
46 CONTINUE
  BA=1.0-RM/RF(J)
47 DO 50 I=1, IT
    IF (P(I)-100000.0) 48,50,50
48 B=0.402*D(1,1)-1.571*D(2,1)+1.972*D(3,1)+0.186*D(4,1)
    DO 49 L=1, 3
      L1=L+1
      D(L,1)=D(L1,1)
49 CONTINUE
    G(I)=G(I)+2.0*BA*B*P(I)/RM
50 CONTINUE

```

```

      BRANCHING PROGRAM

```

```

  Q=Q+BA*BA
  IF (BM*BM-BA) 52,55,55
52 BM=-BA
  JM=J
53 J=J+2
  L=9
  LB=4
  XR=XR*F*F
  X=10.5*XR

```

```

  IF (J-JZ) 19,19,56
56 IF ((J/2)*2-J) 57,58,57
57 XR=XA*F
  J=2
  COTO 18
58 Q=SQRT(Q/FLOAT(JZ))

```

```

      OUTPUT OF INTERMEDIATE RESULTS

```

```

      WRITE(5,142)IC,ST,Q
      INUMB=INUMB+1
      IF (INUMB.EQ.5)COTO 960
      COTO 1100
960 INUMB=0
990 WRITE(5,970)
970 FORMAT(' Continue Iterations? (Y or N) ')
      READ(1,1002)ANS
      WRITE(5,2000)
      IF (ANS.EQ.NNO)COTO 115
      IF (ANS.NE.YES)COTO 990
1100 CONTINUE

```

```

      IF (IM) 60,63,60
60 IF (E-Q) 61,115,115
61 IF (IC-30) 62,115,115
62 IF (E/5.0-ST) 63,115,115
63 IC=IC+1
68 B=0.0

```

```

      COMPUTE THE CHANGES OF THE VALUES OF THE LAYER PARAMETERS
      IN THE FIRST PHASE

```

```

  DO 70 I=1,IT
    G(I)=G(I)/FLOAT(JZ)
    B=B+G(I)*G(I)
70 CONTINUE
  GR=SQRT(B)
  IF (IM) 89,75,89
75 IF (Q-QV) 81,88,88
81 ST=(Q-Q-0.9*E)/GR
  IF (0.5-ST) 83,84,84
83 ST=0.5
84 DO 85 I=1,IT
    GL(I)=G(I)
    PL(I)=P(I)
    PC=ST*G(I)/GR
    P(I)=P(I)*(1.0+PC)
85 CONTINUE
  COTO 100
88 SR=ST

```

```

      COMPUTE THE CHANGES OF THE VALUES OF THE LAYER PARAMETERS
      IN THE SECOND PHASE

```

```

  COTO 91
89 IF (QV-Q) 90,90,91
90 SR=ST
91 B=0.0
  DO 98 I=1,IT
    A=P(I)
    IF (A-100000.0) 92,97,97
92 IF (G(I)/GL(I)) 94,97,93
93 PC=SR*U(I)*G(I)/GR
    P(I)=A*(1.0+PC)
  COTO 95
94 BA=GL(I)/(GL(I)-G(I))
  P(I)=PL(I)+BA*(A-PL(I))
  U(I)=RB*U(I)
  PC=P(I)/A-1.0
95 B=B+PC*PC
  GL(I)=G(I)
97 PL(I)=A
98 CONTINUE
  ST=SQRT(B)
  IF (IM) 99,99,100
99 IM=1
  SR=RA*SR

```

```

100 DO 103 I=1, IK
C
C   ADJUST THE VALUES OF LAYER THICKNESSES
C
  IF (100000.0-P(I)) 101, 101, 103
101 P(I) = 0.0
  DW = 0.0
  DO 102 K=1, I
    DW = DW + PL(K) - P(K)
102 CONTINUE
  P(I) = DW
103 CONTINUE
  QV = Q
  GOTO 16
115 CONTINUE

C
C   OUTPUT THE FINAL RESULTS
C
  DO 120 I=1, IK
    I1K = I + IK
    DW = P(I)
    RW = P(I1K)
    IF (100000.0-DW) 116, 116, 117
116 DW = DW - 100000.0
117 IF (100000.0-RW) 118, 118, 180
118 RW = RW - 100000.0
180 RHO(I) = RW
120 RD(I) = DW
    RW = P(I1K)
    IF (100000.0-RW) 121, 121, 122
121 RW = RW - 100000.0
122 RHO(I1K) = RW
    DEPTH(I) = RD(I)
    DO 170 I=2, IK
      DEPTH(I) = DEPTH(I-1) + RD(I)
170 CONTINUE

C
C   PLOT GRAPH
C
  WRITE (5, 2000)
  CALL J02L03
  CALL MODPLT(ZZB, RMM, JZ, XST, AM)
  CALL RESPLT(OAB, OAR, JZZ, XST, AM)

C
C   WRITE MINI VERSION OF INTERPRETATION
C
  WRITE (5, 1234)
1234 FORMAT(' INTERPRETATION'// ' Thick Rho')
  RD(1) = DEPTH(1)
  IF (IK .LT. 2) GOTO 1235
  DO 1235 IK1=2, IK
    IK1 = IK1 - 1
    RD(IK1) = DEPTH(IK1) - DEPTH(IK1-1)
1235 CONTINUE
1236 WRITE (5, 1237) (RD(IK1), RHO(IK1), IK1=1, IK)
1237 FORMAT(2X, F7.2, F8.1)
  WRITE (5, 1238) RHO(1L)
1238 FORMAT(9X, F8.1)

C
  WRITE (5, 148) Q
  WRITE (5, 149) BM, JM
  WRITE (5, 150) IC
403 WRITE (5, 400)
400 FORMAT(' Do you wish to print the interpretation (Y or N)')
  READ (1, 1002) ANS
  IF (ANS .EQ. RNO) GOTO 402
  IF (ANS .NE. YES) GOTO 403

C
  WRITE (2, 77)
77 FORMAT(1H1)
  WRITE (2, 144) AIP1, AIP2
  CALL PMOD(RHO, RD, DEPTH, REFCO, IK, 1L, 2, OAB, OAR, JZZ)
  WRITE (2, 148) Q
  WRITE (2, 149) BM, JM
  WRITE (2, 150) IC
8881 WRITE (5, 4444)
4444 FORMAT(' Do you wish a print of the model plot? (Y or N)')

```

```

  READ (1, 1002) ANS
  IF (ANS .EQ. RNO) GOTO 402
  IF (ANS .NE. YES) GOTO 8881
  WRITE (2, 6575)
6575 FORMAT(/1X' PLOTTED RESULTS'/)
  CALL EPSON(0, 2)
402 CONTINUE
  JFLG = 1

C
185 WRITE (5, 151)
165 READ (1, 1002) ANS
  IF (ANS .EQ. RNO) GOTO 152
  IF (ANS .NE. YES) GOTO 185
  GOTO 800
152 CALL RESOL(0, 2)
  RETURN
1000 FORMAT(12)
1002 FORMAT(A1)
1001 FORMAT(F10.0)
1010 FORMAT(' Do you want to alter this sounding data? (Y or N)')
132 FORMAT(1X, 'Give acceptable fractional R.M.S. error'// 'to 10 dig
  its including a decimal point')
137 FORMAT(1X, 'Correct? (Y or N)') (C) will cancel the
  'current instruction')
1006 FORMAT(5X, 13, 3X, F6.1, 4X, F10.2)
1007 FORMAT(/ ' V.E.B. No. ', 2A4/)
1003 FORMAT(F10.0)
138 FORMAT(1X, 'Give number of layers (up to 10): ')
139 FORMAT(1X, 'Give the thickness of layer', I2: ')
140 FORMAT(1X, 'Give the resistivity of layer', I2: ')
142 FORMAT(1X, 13, ' Iterations, Steplength =', F7.4, ' R.M.S. error =
  ', F7.4)
144 FORMAT(/ ' V.E.B. No. ', 2A4/)
145 FORMAT(1X, 'Resistivity Thickness')
146 FORMAT(2F10.2)
147 FORMAT(F10.2)
148 FORMAT(/, 1X, 'R.M.S. Relative error =', F8.4)
149 FORMAT(1X, 'Maximum rel. error =', F8.4, ' at sample', I4)
150 FORMAT(1X, 'Number of trials was', I6/)
151 FORMAT(1X, 'Do you want to continue interpreting this sounding?')
  IF (Y or N) //
  END

```

Subroutine EDITXY

The purpose of this subroutine is to edit pairs of data points. They may be deleted, added or altered. The editing is done interactively by the operator through the keyboard using a question and answer procedure. The variables are:

- X - vector containing the x values of the data
- Y - vector containing the y values of the data
- N - the number of data pairs

```

SUBROUTINE EDITXY(X, Y, N)
  DIMENSION X(25), Y(25)
  DATA YES, RNO, CA/ 'Y', 'N', 'C'
22 WRITE (5, 100)
  READ (1, 1002) ANS
  IF (ANS .EQ. RNO) GOTO 20
  IF (ANS .NE. YES) GOTO 22
21 WRITE (5, 102)
  READ (1, 103, ERR=21) J
  IF (J) 22, 23, 24
23 WRITE (5, 108)
  READ (1, 109) XP
  WRITE (5, 111)

```

```

READ(1,109) YP
DO 25 J=1,N
IF (X(J)-XP) 25,26,25
26 IF (Y(J)-YP) 25,24,25
25 CONTINUE
WRITE(5,103) XP,YP
GOTO 21
24 WRITE(5,104) J,X(J),Y(J)
READ(1,1002)ANS
IF(ANS.EQ.RNO)GOTO 21
IF(ANS.EQ.CA)GOTO 22
IF(ANS.NE.YES)GOTO 24
27 N1=N-1
IF (N-J) 21,60,29
29 DO 28 I=J,N1
I1=I+1
X(I1)=X(I1)
28 Y(I1)=Y(I1)
60 CONTINUE
N=N1
GOTO 22
20 WRITE(5,105)
READ(1,1002)ANS
IF(ANS.EQ.RNO)GOTO 30
IF(ANS.NE.YES)GOTO 20
31 WRITE(5,107)
READ(1,105,ERR=31) J
IF (J) 20,32,33
32 WRITE(5,108)
READ(1,109,ERR=32) XP
636 WRITE(5,111)
READ(1,109,ERR=636) YP
DO 34 J=1,N
IF (X(J)-XP) 34,35,34
35 IF (Y(J)-YP) 34,33,34
34 CONTINUE
WRITE(5,103) XP,YP
GOTO 31
33 WRITE(5,104) J,X(J),Y(J)
READ(1,1002)ANS
IF(ANS.EQ.RNO)GOTO 31
IF(ANS.EQ.CA)GOTO 20
IF(ANS.NE.YES)GOTO 33
36 IF (N-J) 31,62,38
38 J1=N-J
DO 37 I=1,J1
N1=N-I+1
NJ=N1+1
X(NJ)=X(N1)
37 Y(NJ)=Y(N1)
62 CONTINUE
N=N+1
J1=J+1
637 WRITE(5,108)
READ(1,109,ERR=637) X(J1)
638 WRITE(5,111)
READ(1,109,ERR=638) Y(J1)
GOTO 20
30 WRITE(5,110)
READ(1,1002)ANS
IF(ANS.EQ.RNO)GOTO 40
IF(ANS.NE.YES)GOTO 30
41 WRITE(5,102)
READ(1,105,ERR=41) J
IF (J) 30,43,44
43 WRITE(5,108)
READ(1,109,ERR=43) XP
639 WRITE(5,111)
READ(1,109,ERR=639) YP
DO 45 J=1,N
IF (X(J)-XP) 45,46,45
46 IF (Y(J)-YP) 45,44,45
45 CONTINUE
WRITE(5,103) XP,YP
GOTO 41
44 WRITE(5,104) J,X(J),Y(J)

```

```

READ(1,1002)ANS
IF(ANS.EQ.RNO)GOTO 41
IF(ANS.EQ.CA)GOTO 30
IF(ANS.NE.YES)GOTO 44
47 WRITE(5,108)
READ(1,109,ERR=47) X(J)
640 WRITE(5,111)
READ(1,109,ERR=640) Y(J)
GOTO 30
40 RETURN
100 FORMAT(' Do you want to remove a point? (Y or N)')
1002 FORMAT(A1)
102 FORMAT(' Give the number of the point ' / ' if you do not know the
number answer 0')
105 FORMAT(I2)
103 FORMAT(' No point with these values has been found ',2F7.2)
104 FORMAT(' Is this the correct point? ',I3,2F8.2/ ' (Y or N)')
1 ' (C) will cancel the current instruction')
106 FORMAT(' Do you want to add a point? (Y or N)')
107 FORMAT(' Give the number of the preceding point' / ' if you do not
know the number answer 0')
108 FORMAT(' Give the electrode spacing of the point')
109 FORMAT(F10.0)
110 FORMAT(' Do you want to change a point? (Y or N)')
111 FORMAT(' Give the apparent resistivity of the point')
END

```

Subroutine CUBSPL

This subroutine calculates the values of a cubic spline for a function. The variables are:

- N - the number of points on the function
- N3 - the size of the working area vector, at least 3 times N
- X - vector containing the x values of the function
- Y - vector containing the y values of the function
- D - vector returned with the values of the spline at the x values of the function
- A - vector used as working space

```

SUBROUTINE CUSP1(N,N3,X,F,D,A)
DIMENSION X(N),F(N),D(N),A(N3)
DO 5 I=2,N
IF (X(I)-X(I-1)) 1,1,5
1 WRITE(5,3) I
3 FORMAT(' RETURN FROM CUSP1 BECAUSE X(',I3,') OUT OF ORDER')
A(I)=1.0
RETURN
5 CONTINUE
I=0
30 I=I+1
J=2
IF (I-1) 6,10,6
6 J=N-1
IF (I.EQ.N) GOTO 10
I1=I-1
H1=1.0/(X(I)-X(I1))
I1=I+1
H2=1.0/(X(I)-X(I1))
ITEMP=3*I-2
A(ITEMP)=H1
ITEMP=3*I-1
A(ITEMP)=2.0*(H1+H2)
ITEMP=3*I
A(ITEMP)=H2

```

```

TEMP=(H1-H2)/H1-H2
I1=1
I1=1
D(1)=3.0*(F(I1)*H2*H2+F(I1)*TEMP-F(I1)*H1*H1)
GOTO 30
10 J1=J-1
H1=1.0/(X(J)-X(J1))
J2=J+1
H2=1.0/(X(J)-X(J2))
ITEMP=3*I-2
A(ITEMP)=H1-H2
ITEMP=3*I-1
A(ITEMP)=H1-H2*H2
ITEMP=3*I
A(ITEMP)=H1-H2*H2
TEMP=H1-H2*H2
TEMP1=H2-H2*H2
D(1)=2.0*(F(J1)*TEMP1-F(J2)*TEMP1-F(J1)*TEMP)
IF (I.LT.N) GOTO 30
P=A(4)/A(1)
A(5)=A(5)-P*A(2)
A(6)=A(6)-P*A(3)
D(2)=D(2)-P*D(1)
I=2
50 I=I+1
K=3*I-4
ITEMP=K+2
P=A(ITEMP)/A(K)
ITEMP=K+3
ITEMP1=K+1
A(ITEMP)=A(ITEMP1)-P*A(ITEMP1)
ITEMP=I-1
D(1)=D(1)-P*D(ITEMP)
IF (I.NE.N-1) GOTO 150
ITEMP=K+5
P=A(ITEMP)/A(K)
ITEMP1=K+6
ITEMP2=K+1
A(ITEMP)=A(ITEMP1)-P*A(ITEMP2)
ITEMP=K+7
A(ITEMP)=A(ITEMP)
ITEMP=N-2
D(N)=D(N)-P*D(ITEMP)
150 IF (I.LT.N) GOTO 50
ITEMP=3*N-1
D(N)=D(N)/A(ITEMP)
I=2
60 I=I+1
J=N+2-I
ITEMP=3*J
ITEMP1=J+1
TEMP=(D(J)-A(ITEMP)*D(ITEMP1))
ITEMP=3*J-1
D(J)=TEMP/A(ITEMP)
IF (I.LT.N) GOTO 60
D(1)=(D(1)-D(2)*A(2)-D(3)*A(3))/A(1)
A(1)=0.0
RETURN
END

```

Function CUSP2

The purpose of this function is to evaluate a spline. The input and output variables are:

IX - a flag set to 0 on first entry of the function and -1 subsequently
 N - the number of points on the function
 U - vector containing the x values of the function
 S - vector containing the y values of the function

D - vector returned with the values of the spline at the x values of the function
 X - x value of the point at which the spline is to be evaluated

```

FUNCTION CUSP2(IX,N,U,S,D,X)
DIMENSION U(N),S(N),D(N)
IFLG=0
IEPS=.19
IF (X.LT.U(1)) GOTO 990
IF (X.GT.U(N)) GOTO 991
IF (IX.LT.0.OR.IFLG.EQ.0) GOTO 12
J1=J+1
IF (X.LE.U(J1)) GOTO 8
J=J+1
IF (J-25) 11,30,30
30 WRITE(5,80)
80 FORMAT(' J=25 ')
GOTO 7
11 J1=J+1
IF (X-0.1) 21,21,1
31 GOTO 7
12 J=ABS(X-U(1))/(U(N)-U(1))*(N-1)+1
ITEMP=N-1
J=MIN0(J,ITEMP)
IFLG=1
IF (X.GE.U(J)) GOTO 11
2 J=J-1
IF (J.GT.0) GOTO 20
WRITE(5,81)
81 FORMAT(' J=0 ')
GOTO 7
20 IF (X.LT.U(J)) GOTO 2
J1=J+1
H=U(J1)-U(J)
Q1=H*D(1)
Q2=H*D(J1)
SS=S(J1)-S(J)
B=3.0*SS-2.0*Q1-Q2
A=Q1-Q2-2.0*SS
8 Z=(X-U(J))/H
CUSP2=((A*Z+B)*Z+Q1)*Z+S(J)
RETURN
990 TEMP=AMAX1(ABS(U(1)),ABS(U(N)))
IF (X-(U(1)+2.0*IEPS*TEMP)) 99,99,33
33 J=1
GOTO 7
991 TEMP=AMAX1(ABS(U(1)),ABS(U(N)))
IF (X-(U(N)+2.0*IEPS*TEMP)) 34,99,99
34 J=N-1
GOTO 7
99 IFLG=0
CUSP2=0.0
RETURN
END

```

Subroutine PMOD

The purpose of this subroutine is to list a geoelectrical layered model and an apparent resistivity curve on the printer. The input variables are:

RHO - vector containing the layer resistivities
 RD - vector containing the layer thicknesses
 DEPTH - vector containing the layer boundary depths
 REFCO - vector containing the reflection coefficients
 NMI - the number of layers -1
 NLAY - the number of layers

NC - the output channel number
 OAB - vector containing the apparent resistivity curve
 electrode spacings
 OAR - vector containing the apparent resistivity curve
 resistivity values
 JZZ - the number of points on the apparent resistivity curve

```

SUBROUTINE PMOD(RHO, RD, DEPTH, REFCO, NMI, NLAY, NC, OAB, OAR, JZZ)
  DIMENSION RHO(NLAY), REFCO(NMI), RD(NMI), DEPTH(NMI)
  DIMENSION OAB(25), OAR(25)
  DO 3 I=1, NMI
    I1=I+1
    REFCO(I) = (RHO(I1)-RHO(I))/(RHO(I1)+RHO(I))
  3 CONTINUE
  RD(1)=DEPTH(1)
  IF (NMI-2) 2,6,6
  6 DO 4 I=2, NMI
    I1=I-1
    RD(I)=DEPTH(I)-DEPTH(I1)
  4 CONTINUE
  2 WRITE(NC, 1040)
  NLAY4=NLAY+4
  DO 5 I=1, NLAY
    J1=I+3
    J2=J1+1
    J3=J2+1
    IF(J1.LE. JZZ)WRITE(NC, 1050)J1, OAB(J1), OAR(J1)
    IF(J1.GT. JZZ)WRITE(NC, 1080)
    IF(I.LT. NLAY)WRITE(NC, 1060)RD(I), RHO(I)
    IF(I.EQ. NLAY)WRITE(NC, 1055)RHO(I)
    IF(J1.LE. JZZ)WRITE(NC, 1050)J1, OAB(J1), OAR(J1)
    IF(J1.GT. JZZ)WRITE(NC, 1080)
    IF(J2.LE. JZZ)WRITE(NC, 1050)J2, OAB(J2), OAR(J2)
    IF(J2.GT. JZZ)WRITE(NC, 1080)
    IF(I.LT. NLAY)WRITE(NC, 1070)DEPTH(I), REFCO(I)
    IF(I.EQ. NLAY)WRITE(NC, 1065)
    IF(J3.LE. JZZ)WRITE(NC, 1050)J3, OAB(J3), OAR(J3)
    IF(J3.GT. JZZ)WRITE(NC, 1080)
  5 CONTINUE
  IF(JZZ.LE. NLAY4)GOTO 99
  JAA=NLAY4+1
  WRITE(NC, 1050)(I, OAB(I), OAR(I), I=JAA, JZZ)
  1080 FORMAT(1H )
  1050 FORMAT(5X, 13, 3X, F6.1, 4X, F10.2)
  1060 FORMAT(1H+, 39X, F7.2, 12X, F8.1)
  1070 FORMAT(1H+, 47X, F8.2, 3X, 8(1H-), 3X, F7.4)
  1055 FORMAT(1H+, 58X, F8.1)
  1065 FORMAT(1H+, 58X, 8(1H+))
  1040 FORMAT(/ / 13X, 'FIELD CURVE DATA', 23X, 'INTERPRETED MODEL' / /,
    21X, 'Electrode', 4X, 'Apparent', 39X, 'Reflection',
    3/10X, 'Separation', 3X, 'Resistivity', 6X, 'Thickness', 2X,
    4'Depth', 6X, 'Rho', 7X, 'Coeffts. / /)
  99 RETURN
  END

```

Subroutine MODPLT

This subroutine plots a continuous apparent resistivity curve that has been calculated from the layered model. The curve values are interpolated from the input values using the spline routines. The input variables are:

XX - vector containing the electrode separations
 Y - vector containing the apparent resistivity values

N - the number of points on the apparent resistivity curve
 XST - the value of the leftmost log cycle in the x-axis for plotting
 the curve on
 AM - the logarithm of the mean apparent resistivity value

```

SUBROUTINE MODPLT(XX, Y, N, XST, AM)
  DIMENSION XX(N), Y(N), C(25), W(75), X(25)
  DO 20 I=1, N
    X(I)=80.0*(ALOG10(XX(I))-XST)+79
    Y(I)=80.0*(ALOG10(Y(I))-AM)
  20 CONTINUE
  N3=N+3
  CALL CUBP1(N, N3, X, Y, C, W)
  IF (W(1)) 21, 22, 21
  22 IE=-1
  IX=INT(X(1))
  IXX=INT(X(N))
  DO 23 J=IX, IXX
    AXX=FLOAT(J)
    IY=INT(CUBP2(IE, N, Y, C, AXX))
    IF (IY) 24, 25, 25
  24 IY=IY+160
  25 IF (IY-160) 27, 27, 28
  28 IY=IY+160
  27 IY=IY+31
  IE=1
  CALL PLOT(J, IY, 2)
  23 CONTINUE
  RETURN
  21 STOP
  END

```

REFERENCES

- Barker, R.D., 1981. "The Offset System of Electrical Resistivity Sounding and its Use with a Multicore Cable", Geophysical Prospecting vol. 29, pp 128-144.
 Ghosh, D.P., 1971. "Inverse Filter Coefficients for the Computation of Apparent Resistivity Standard Curves for a Horizontally Stratified Earth", Geophysical Prospecting vol. 19, pp 769-775.
 Koefoed, O., 1979a. "Geosounding Principles, 1, Resistivity Sounding Measurements", publ. Elsevier, Netherlands.
 Koefoed, O., 1979b. "Adaption of the Method of Steepest Descent to Resistivity Sounding Interpretation" paper presented at 41st meeting of EAEG at Hamburg.